ST-STE-TT
AMIGA
MAC et PC
émulation



Le magazine
des 16/32 bits

N° 49 / 25F
Fév/Mars 91

TT :
MÉMOIRE
ET
PERFORMANCES

ST

CARTE CHILI : ENFIN DU MULTIMEDIA SUR ST !
GÉRER SON BUDGET : 3 NOUVEAUX OUTILS
ÉDUCATIFS : UNE NOUVELLE VAGUE
PROGRAMMER LE SON DU STE
LE DESSIN TECHNIQUE
LE RÉDACTEUR 3.10
DIDOT LINEART

MUSIC MASTER :
L'ATELIER MUSICAL

AMIGA

UNE TABLETTE GRAPHIQUE
LE FORMAT IFF : SUITE !
PAINTER 3D



BELGIQUE : 180 FB CANADA : 6.95$C SUISSE : 7.50 FS

GÉNÉRATION 4

AMIGA - ATARI ST - PC - CONSOLES

SPECIAL C.E.S DE LAS VEGAS: 30 PAGES DE NEWS INCROYABLES!

PREVIEWS:

Alcatraz
Might & Magic 3
Super Monaco G-Prix
Switchblade 2
Wing Commander 2
World Of Ultima: 2

TESTS:

Lemmings
Obitus
Red Barron
Rise Of The Dragon
Team Suzuki
Turrican 2



L'HISTOIRE SANS FIN 2
le film, le jeu

CONSOLES: TOUTES LES NEWS ET PREVIEWS POUR LYNX, GAME-BOY, GAME-GEAR, COREGRAFX, NEO-GEO, MEGADRIVE & SUPER-FAMICOM

SOYEZ PRÉSENTS À LA GRANDE SOIRÉE DES 4 D'OR

EDITO

FAISONS LE POINT...

Un certain nombre d'indicateurs, en cette fin Janvier, nous poussent à quelques considérations prospectives...

D'une part, à l'heure où nous bouclons les dernières pages, voici qu'enfin nous disposons d'informations assez précises mais non encore exhaustives sur ce fameux MegaSTE, qui a fait son apparition publique en Allemagne et en Hollande. A tel point que nous avons quelque peu "compacté" nos News, afin de vous livrer le plus tôt possible le maximum d'éléments sur cette réplique d'Atari à l'évolution actuelle des différentes offres du marché, et que nous approfondirons dès le prochain numéro, après le Forum PC où il sera officiellement présenté au public. Incontestablement, cette "nouvelle" machine présente des caractéristiques et un coût prévisionnel qui la placent intelligemment dans la gamme des micros accessibles au plus large public, en lui évitant d'être considérée comme une simple "mise à jour" plus ou moins opportuniste.

D'autre part, l'arrivée de la carte Chili consacre ce que l'on pourrait appeler l'accession du ST à l'univers du "multimedia". Essentiellement composé, pour l'instant, d'une liaison avec le monde de la vidéo - c'est pourquoi nous en avons profité pour démarrer une initiation à la vidéo, répondant ainsi à une demande de plus en plus pressante d'un grand nombre de lecteurs -, cet univers passionnant devient enfin une réalité pour une machine dont les limitations originelles ne lui laissaient que peu d'espoirs d'offrir à son public les moyens de se livrer à ce type d'activité. Certes, les moyens budgétaires à mettre en oeuvre ne sont sans doute pas encore à la portée de tout un chacun, mais les progrès sont tangibles et la baisse évidente, depuis quelques années, des coûts des matériels dans ce domaine, laisse présager une véritable explosion de la créativité personnelle à l'aide de son micro-ordinateur.

Enfin, la "professionnalisation" et la puissance des outils logiciels en différents domaines sur ST vient corroborer cette évolution, et ST Magazine se fera un devoir de suivre pour vous, sinon d'anticiper, les ouvertures multiples vers cette créativité personnelle.

Directeur de la publication : Godefroy Giudicelli. Rédacteur en chef : François Gabert. Rédacteurs en chef adjoints : Stéphane Moreau et Jacques Caron. Chefs de Rubriques : Laurent Katz, Frank Ladoire, François Pagès, Daniel Fournier, Sébastien Mougey, Henri Abdelouab.
Secrétaire de rédaction : Françoise Germain.
Fabrication, maquette et photogravure : Michel Lhopitault, Mireille Guérineau. Rédacteurs graphiques PAO : Olivier Ferfache, Jean Minthe. Photographe : François Paupert.
3615 STMAG: Mic Dax, STJC.
Abonnements : Nicole Gabert. Services Comptables : Charles Convalot, Alain Tatem, Claudine Clément.
Ont participé à ce numéro : Guillaume Saviard.
Special dedicace to CNN. Anti-dedicace to CNN.
ST Magazine est une publication Pressimage, SARL de presse au capital de 2000 francs et dont le siège social est situé au 210 rue du Faubourg Saint Martin, 75010 Paris. A ne pas confondre avec l'adresse de la Rédaction, qui attend vos courriers et vos abonnements au 19 rue Hégésippe Moreau, 75018 Paris.
Imprimeurs : SNIL (Aulnay sous bois), RBI (Tournan). Transcodage & photocomposition : Pressimage. Dépôt légal: à parution. Photogravure couleur: G.Y.A (Paris), Prephix (Paris).
Service Promotion : Option Presse Diffusion, terminal E 13. ISSN : 0980-5338. Commission Paritaire : 71845.
Membre inscrit OJD.

Direction de la Publicité : Antoine Harmel. Chef de publicité : Véronique Perrin aux (1) 43.87.01.39 et (1) 45.22.38.60.
Gestion commerciale : Jérôme Forneris



SOMMAIRE

TÉLÉMATIQUE

MATOS

LANGAGES

APPLICATIONS VERTICALES

TECHNIQUES DE L'IMAGE

CAHIER AMIGA

Index des Annonceurs

LE MEGA STE !

Nous vous avions déjà donné quelques éléments sur cette nouvelle machine qu'Atari a présenté au Comdex d'automne. Maintenant qu'elle arrive en Europe (elle serait même disponible dans certains pays, en particulier les Pays-Bas) et même en France (présentation lors du PC Forum qui a lieu ces jours-ci, du 12 au 15 février), on commence à un savoir un peu plus.

Voici donc les caractéristiques définitives de la machine :
- microprocesseur 68000 à 16 MHz, passable à 8 MHz ;
- coprocesseur arithmétique optionnel 68881 à 16 MHz ;
- cache de 16 ko, désactivable ;
- 1, 2 ou 4 Mo de RAM (barrettes SIMM) suivant versions ;
- Bus VME (pas de Bus Mega ST !) ;
- 1 port réseau de type LocalTalk et 2 ports séries contrôlés par un Zilog 8350 (comme sur TT) ;
- disque dur intégré facilement changeable ;
- 256 ko de ROM avec le TOS 2.5 ;

Tout le reste (modes graphiques, son, Blitter...) est strictement identique au STE, à part les ports joystick 15 broches qui ont disparu (c'était déjà le cas sur TT).

La machine se présente dans un boîtier identique à celui du TT, mis à part sa couleur, qui tend vers le gris foncé, alors que le TT est blanc cassé. Il en est de même pour le clavier, avec toujours les prises DB9 de connexion de joystick ou souris sur le bord du clavier, faisant qu'on perd encore un peu de place sur son espace de travail. Le boîtier, comme sur TT, est composé de deux parties clairement séparables. Au dessus de la base (dont toute la surface est occupée par la carte-mère, la première partie, à peu près carrée, contient le lecteur de disquettes, l'alimentation, et l'emplacement pour le bus VME.

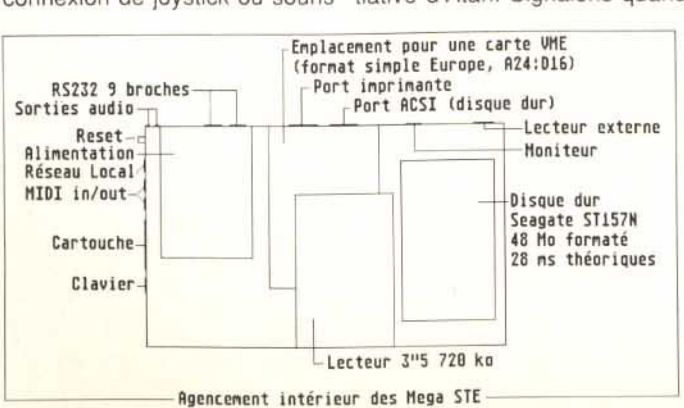
La deuxième partie, sur la droite, est facilement amovible (une simple vis à dévisser), et contient le disque dur (fixé sur le boîtier). Une fois ouvert, on trouve directement en dessous les 4 supports de barrettes SIMM (placés à 45 degrés pour gagner de la place, vu qu'il n'en reste pas beaucoup sous le disque dur), ainsi que les 2 ROMs contenant les 256 ko du TOS, et la rangée de 8 micro-interrupteurs qui avaient fait leur apparition sur TT. C'est une excellente initiative d'Atari que d'avoir placé tous ces composants fréquemment changés à un endroit aussi accessible !

De toutes façons, le reste de la machine n'est pas formidablement inaccessible. Il n'y a que quelques vis à enlever pour découvrir la carte mère : pas le moindre blindage en vue, le boîtier étant simplement recouvert d'une fine couche d'un produit de couleur rouille qui est censé jouer le même rôle. C'est encore une fois une très bonne initiative d'Atari. Signalons quand même, que si le démontage est facile, il n'en annule pas moins la garantie (comme Atari en a l'habitude depuis quelque temps, il y a un sceau de garantie sur l'une des vis).

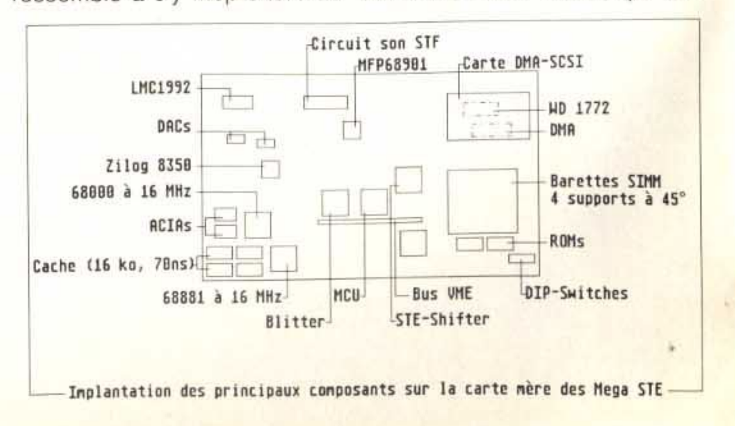
Vous trouverez sur les 2 figures illustrant cet article l'implémentation des principaux composants, ainsi que l'agencement intérieur et la disposition des connecteurs d'entrée-sortie. Ce ne sont que des schémas approximatifs, réalisés en toute hâte d'après des photos parues dans "ST-Nieuws", un magazine néerlandais, fort bien informé, dont émane d'ailleurs une majeure partie des informations ici présentes (aux erreurs de traduction près, heureusement que ça ressemble à de l'allemand...).

A propos des interfaces, 3 remarques : d'abord, les connecteurs série sont, comme sur TT, des prises 9 broches ; ensuite, l'un des 3 connecteurs (non représenté sur la figure) est situé sur la plaque de protection arrière du bus VME, comme sur TT, et devra donc être oublié (ou reporté un peu plus loin à l'aide de quelques décimètres de câble en nappe) en cas d'utilisation dudit bus. Pour finir, il n'y a pas de bus SCSI apparent à l'extérieur de la machine, mais le disque dur interne est SCSI, et connecté par l'intermédiaire de la carte DMA-SCSI d'Atari, il serait donc possible d'y "récupérer" le bus SCSI sans difficulté.
Au niveau logiciel, le Mega STE dispose donc du TOS 2.05, qui ressemble à s'y méprendre au 3.01 du TT (amusant... Atari a toujours utilisé des versions paires (1.0, 1.2, 1.4, 1.6), et tout à coup, on n'a plus que des versions impaires !), en reprenant surtout le nouveau bureau (les ROMs correspondantes devraient être disponibles prochainement pour les possesseurs de STE, les STF et Mega n'ayant droit, semble-t-il, qu'au TOS 1.4, enfin disponible).
Le nouveau panneau de contrôle est fourni, avec des nouvelles versions des quelques CPX (modules du panneau de contrôle) qui ont pu changer, essentiellement "Général", où on trouve maintenant la configuration de la vitesse du 68000 et du cache, laissant le choix entre 8 MHz sans cache, 16 MHz sans cache, et 16 MHz avec cache, les deux premières configurations étant prévues pour les logiciels qui ne supporteraient pas soit le cache, soit la fréquence plus rapide.
Les utilitaires de disque dur (HDX, AHDI...) en sont maintenant à la version 4.02, mais nous n'en savons pas plus sur les éventuelles nouveautés (il s'agit probablement essentiellement d'une adaptation au Mega STE, sans rien de plus).

Vous avez maintenant une description que nous espérons fidèle de la machine, du point de vue théorique. Voyons maintenant les performances qui ont pu être mesurées sur la machine, à l'aide de Quick Index, qui est



Agencement intérieur des Mega STE



Implantation des principaux composants sur la carte mère des Mega STE

vraiment le test universel (voyez l'article sur les performances du TT dans ce même numéro).
En voici les résultats :

Mode	8 MHz ss cache	16 MHz av. cache	16 MHZ ss cache
CPU memory	100	100	164
CPU register	100	100	204
CPU divide	100	181	203
CPU shifts	100	178	207
BIOS text	107	111	119
BIOS string	104	108	122
BIOS scroll	132	133	134
GEM draw	142	146	171

On voit donc qu'au niveau des performances "brutes" (les 4 premiers tests), le Mega STE va environ deux fois plus vite sur la plupart des opérations, n'étant freiné que par l'accès à la RAM, pour lequel on ne gagne que 60%.
En ce qui concerne les tests du TOS, c'est beaucoup plus subtil. En effet, si on compare les résultats obtenus avec ceux du ST (100% dans tous les tests), on constate un gain de l'ordre de 20% à 30% sur le BIOS, et plus de 70 sur le GEM, ce qui semble particulièrement encourageant... Mais si on consulte la colonne "8 MHz", on constate qu'au lieu de trouver 100% dans tous les cas, on obtient déjà des gains relativement importants (faibles si on compare à Turbo ST, mais le TOS a l'avantage d'être moins sensible aux problèmes de résolutions "non standard"). Le gain dû au processeur est donc relativement faible, semble-t-il, pour le TOS, mais le résultat final est encourageant (il existe une autre possibilité, qui serait que les ROMs contenant le TOS soient plus rapides d'accès, mais je préfère imaginer qu'Atari s'est un peu démené pour accélérer le TOS).
ST-Nieuws a aussi effectué un certain nombre de tests "pratiques" sur le Mega STE. Les voici, avec à chaque fois les résultats sur ST, Mega STE, et TT.

- Affichage d'une page complexe (avec beaucoup de bitmaps) sous Calamus : 9 secondes, 5 secondes, 2,3 secondes.
- Projection d'un objet sur un "bol bleek" (vous parlez néerlandais, vous ? Pas moi), le Mega STE est 176% plus rapide que le ST, et le TT 411%.
- Compilation d'un programme comprenant de nombreux fichiers. Sur disque dur : 237, 154 et 117 secondes respectivement. Avec un RAM-disque, 145, 105 et 32 secondes (le cas du disque dur est à prendre avec beaucoup de précautions, à cause des différences de disque d'une part, et du fait que le TOS a été largement accéléré de ce point de vue à partir du 1.4).

On se rend compte que le Mega STE est donc relativement rapide, ce qui satisfera nombre d'utilisateurs, qui n'ont pas pour autant les moyens d'acheter un TT. D'ailleurs, le Mega STE a un avantage sur le TT, outre son prix, qui est une compatibilité probablement bien plus importante, la seule grosse différence par rapport au ST du point du vue hard étant la fréquence du processeur et le cache, mais qu'on peut facilement configurer pour simuler exactement un ST. Bref, le seul problème ne peut venir que des programmes qui ne respectent pas les règles de programmation élémentaires, et qui souffriront donc du passage au TOS 2.5. Heureusement, étant donné le nombre de changements de TOS déjà effectués à l'heure actuelle, les développeurs commencent à comprendre les dangers de ce style de programmation.

Voilà donc notre tour fini. Il ne reste plus qu'à espérer la disponibilité rapide de cette machine étonnante, qui lève une bonne partie des défauts des ST et STE, aussi bien au niveau du hard (microprocesseur plus rapide, cache...) que de la réalisation (accès facile aux RAMs, ROMs et disque dur, absence du blindage habituel) et du logiciel (TOS 2.5 avec le nouveau bureau). A l'heure où nous écrivons ces lignes, les configurations disponibles en France et leurs prix n'ont pas encore été fixés, mais nous nous permettons de vous indiquer que le Mega STE 4, avec 4 Mo de RAM (logique !), le disque dur interne de 48 Mo, et le moniteur monochrome SM124, est fixé à 3629 florins TTC aux Pays-Bas, soit moins de 11000 francs ! On parle même en France de prix inférieur à 10000 francs, mais il s'agit peut-être de la version 2 Mo, ou encore de la configuration sans moniteur. L'avenir (et Atari au Forum PC) nous le dira.

CALLIGRAPHER 2.12

On annonce la nouvelle version de Calligrapher (en déclinaisons Pro et junior), prévues en Français, en Allemand et en Espagnol. Avec une dernière version en Italien, Calligrapher prépare l'Europe.

La prochaine version 2.12 comportera l'affichage vectoriel des fontes Gdos / LineArc jusqu'à 128 points (2 Mo de mémoire minimum), sa vitesse d'importation des graphiques se voit multipliée par 5, la vitesse d'affichage de la prévisualisation en "wysiwyg" par 10, et l'on peut exporter ses fichiers au format .GEM, avec le driver Metafile. Cela signifie que l'on pourra recharger ses documents dans les logiciels de PAO acceptant ce format, et même de les convertir en fichiers PostScript (par l'entremise de PPM). Signalons aussi qu'il est désormais possible de travailler avec DEUX imprimantes (une laser et une matricielle par exemple), avec choix par la touche Control. Enfin, on trouve deux nouvelles fonctions mathématiques pour l'édition de formules, et l'essentiel des bugs répertoriés a été supprimé. Dernière news pour la fine bouche : il existe désormais une version TT, en résolutions ST.

Tous les utilisateurs dûment répertoriés bénéficieront d'un courrier annonçant les détails de la procédure de mise à jour. Un bulletin, "Cal News", leur sera également transmis : une mine de trucs et astuces sous Calligrapher ; les utilisateurs Calligraphes et Callipiges sont vivement invités à envoyer leurs propres découvertes et astuces à l'adresse suivante : Upgrade Editions, 28 rue de Coriolis, 75012 Paris.

SCRIPT II

Non, ce n'est pas l'Arlésienne, ni son fils, Script II existe bel et bien mais l'annonce officielle de sa sortie était quelque peu prématurée du fait du très important travail dû à la francisation du logiciel et bien sûr, de son important dictionnaire. Le bout du tunnel approche, et désormais l'échéance de fin Février/ début Mars est annoncée. Application Systems nous annonce aussi que Script II sera accompagné d'un programme indépendant de "Touches Mortes", module qui pourra ensuite être acquis séparément, notamment pour les utilisateurs de Signum. Enfin, tous les utilisateurs répertoriés (ayant renvoyé leurs cartes d'enregistrement) seront personnellement informés par mailing de la disponibilité du produit.

X WINDOWS ET OSF MOTIF PORTÉS SUR NEXT

La firme Pencom Software a récemment annoncé son intention de porter le système de fenêtrage graphique en réseau X-Windows 11.4 et la boîte à outils OSF Motif 1.1 sur la gamme de stations de travail NeXT. Le tout tournera en utilisant les primitives graphiques de NeXTStep, l'intégrateur graphique livré avec les NeXT et basé sur Display PostScript. Si les performances sont correctes, cela devrait donner aux machines de NeXT un atout qui leur manquait encore, celui de pouvoir exécuter des applications graphiques dans un environnement hétérogène.

ZZ-VOLUME

Voici les nouveautés de la version 1.56 :

- lancement direct (sans quitter le programme) d'une autre application (à concurrence de la mémoire disponible) ;
- choix de la quantité mémoire à laisser à disposition pour lancer l'application (ST) et choix du type de mémoire de travail (TT);
- Export Postscript et EPSF (Encapsulated Postscript) en vues filaires ou en faces cachées ombrées ;
- nouvelle commande de mesure d'angle ;
- sélection des entités par attributs (calques, zone, couleur, fiche technique, huisserie, nom d'objet) ;
- affichage des objets global, ou tenant compte des calques;
- export de coupes horizontales de hauteurs variables pour la création automatique de plans 2D ;
- lancement auto du programme par clic sur fichier .V3D ;
- gère 255 calques, nommables, (configuration sauvegardable sur disque) ;
- prémasquage possible de faces et d'arêtes lors de calculs de faces cachées.

LES RETOMBÉES DE L'IDS

L'Initiative de Défense Stratégique chère à Reagan, si critiquée en son temps, a tout de même produit des milliers de retombées technologiques.

Une base de données de 1400 de ces technologies nouvelles, non classifiées, la plupart dans le domaine de l'informatique ou des lasers, a été publiée par le ministère de la Défense américain, en raison de leur possibilité d'usage commercial. On va probablement voir fleurir quelques "start-up" californiennes pour exploiter ces technologies.

L'ELECTRONICIEN VERSION 1.2

Pour ceux qui ignorent tout de l'Électronicien, nous vous renvoyons au banc d'essai sur la version 1.1 paru dans le numéro 44 de ST-MAG (page 28). En attendant que votre copain vous rende l'exemplaire que vous n'auriez jamais dû lui prêter, rappelons que l'Électronicien est le seul simulateur de circuits électroniques analogiques diffusé en France sur ST.

Une nouvelle version de ce progiciel est arrivée avec la nouvelle année : la version 1.2. Elle nécessite 1 mégaoctet de RAM mais apporte des améliorations significatives.

Le problème du tracé des courbes pouvant apparaître pour certains sauts de phase a été résolu. Quelques autres problèmes (mineurs) ont également été corrigés (ergonomie, présentation, impression).

Les boîtes de dialogue sont configurables en mode "Expert" ou "Débutant". En mode "Expert", un simple "Return" confirme votre choix alors qu'en mode "Débutant" vous devez le confirmer à l'aide de la souris.

Les audiophiles ont été entendus : il est désormais possible de définir des plages de fréquences de 20 Hz à 20 kHz (ou n'importe quelle autre plage de fréquences de même type).

Lors du chargement d'un circuit, la simulation peut être exécutée immédiatement ou reportée au moment opportun.

Les calculs peuvent être suspendus en appuyant sur la touche "Escape" et sont menés avec une plus grande précision (8 octets au lieu de 6).

L'amélioration la plus intéressante est incontestablement une diminution de plus de 35 % des temps de calcul : la vitesse est maintenant plus que confortable.

Pour obtenir la dernière version de l'Électronicien, il suffit de renvoyer à LOGISOFT votre disquette originale accompagnée d'un chèque de 60 F.

Quant à la version professionnelle 2.0, elle commence à se faire attendre : au dire de l'auteur, il faudra sans doute patienter jusqu'à Noël...

BUSH SOUTIENDRA LE TÉLÉTRAVAIL

Un nombre croissant de personnes utilisent l'informatique dans leur travail et passent des journées entières devant leur écran, recevant toutes leurs informations par les réseaux informatiques de leur entreprise, et produisant des documents traités ensuite par ces mêmes réseaux. Dans ces conditions, il y a des jours où elles pourraient s'abstenir de venir au travail et se contenter de se connecter par modem aux ordinateurs de l'entreprise.

On s'attend à ce que le président Bush fasse bientôt une déclaration en faveur de cette forme de travail à domicile, nommée télétravail chez nous : le mot vient de "telecommuting" (télécommunications en anglais) et de "commuting" (emprunter les correspondances). Arguments en faveur du télétravail : gain de temps en transports, flexibilité des horaires, confort domestique, et diminution du trafic et de la pollution. D'après les Services Généraux de l'Administration américaine (GSA), le télétravail demeure concevable pendant quelques jours par semaine pour les développeurs, rédacteurs de documents et agents administratifs qui traitent des informations sans besoin de communication verbale. A déconseiller par contre aux superviseurs et... aux mères dont les enfants requièrent l'attention si elles tentent d'être productives chez elles !

En France, le développement du réseau Numéris (56 kbits/s) rend le télétravail envisageable dans beaucoup de domaines. Le principal obstacle sera les syndicats, qui craignent que le télétravail, en assouplissant les tâches, ne démobilise les employés. Et puis, où va-t-on s'il faut utiliser le courrier électronique pour distribuer des tracts aux employés ?

DU YAOURT DANS UNIX, DU RAP DANS PICK ET UN TIGRE DANS VOTRE MICRO !

Ces Américains ne savent plus quoi inventer ! Lors de la dernière Unix Expo, où les exposants rivalisaient de tape-à-l'oeil, la firme Convergent Solutions offrait aux visiteurs de son stand un "frozen yogurt" (une crème glacée au yaourt) servi dans des coupes représentant des casquettes d'équipes de base-ball. L'année prochaine, ils promettent de fournir de la sauce chocolat. Un moyen comme un autre de promouvoir les outils de développement vendus par Convergent.

Par contre, lors du récent Comdex, la firme Pick Systems a choisi un trio de rap et une ravissante pin-up pour attirer l'attention sur le système d'exploitation Pick (qui se pose en complément d'Unix pour les bases de données). Les chansons mises en musique rap ont même été écrites spécialement pour Dick Pick, inventeur du système et président de la firme. Mais les spectateurs ont surtout été attirés par la blonde et ravissante Karla Kiaura, qui pose pour la firme de maillots de bain Body Glove (un client de Pick Systems) et qui dédicaçait ses photos sur le stand Pick, en chair et en formes (NDLR : You know what ? I'm Happy !).

Quant à John Imlay, P.D.G. de Dun & Bradstreet Software (une société de logiciels américaine), il a carrément amené un tigre (le symbole publicitaire de la maison) à une conférence ! Le "minou", de mauvais poil ce soir-là, a quelque peu amoché un participant. Un coup à se retrouver bagnard en costume rayé...

La future station Commodore 3000 UX contre le futur Atari TT/X :
Commodore présentera sa nouvelle station de travail 3000 UX lors de l'UniForum de janvier. Cette machine est architecturée autour d'un 68030 cadencé à 25 MHz et pourra faire tourner le System V.4 d'AT&T ou AmigaDOS (mais pourra-t-on faire tourner les deux en même temps, sous forme de processus séparés, ou devra-t-on relancer la machine pour changer de système d'exploitation ?). La machine sera livrée avec X-Windows et l'interface utilisateur Open Look. Deux versions de la machine sont prévues : la 3000 UX-100 comportera 4 Mo de RAM et un disque de 100 Mo, et la 3000 UX-200 aura 8 Mo de RAM et un disque de 200 Mo. Le prix de la version 100 devrait être de l'ordre de 5000 dollars avec un moniteur couleur de 36 cm. L'université de Virginia Tech a d'ores et déjà abandonné ses Mac II sous A/UX (l'Unix d'Apple) au profit de machines Commodore. Mais le manque d'applications portées sous System V.4 risque de freiner le succès de la machine, qui se pose en concurrent direct de l'Atari TT/X en Europe (lequel souffre du même problème).

Aux USA, Atari traîne à ses basques une image de marque si déplorable que les acheteurs de stations de travail bas de gamme ne mentionnent même pas cette machine dans leurs choix possibles. Espérons pour Atari que l'Europe lui sera plus clémente... si le TT/X devient un jour disponible.

L'AUTEUR D'UN VIRUS BIENTOT INCULPÉ

La police de Dallas, au Texas, réunit les preuves pour inculper l'auteur du virus Scores sur Macintosh. Son nom n'a pas été divulgué, mais il s'agirait d'un ancien employé de la firme Electronic Data Systems (EDS). Pas très malin, ce saboteur a testé son virus sur ses collègues, qui utilisent une application propre à EDS qui n'a jamais de contact avec l'extérieur. Pas difficile de découvrir la source du mal.

LA LÉGION DES (CON)DAMNÉS

Trois membres d'un groupe de "hackers" américains se nommant "La Légion des Damnés" viennent d'être lourdement condamnés à des peines allant de 14 à 21 mois de prison, ainsi qu'à 30.000 dollars de dommages et intérêts chacun. Les bidouilleurs, âgés de 22, 23 et 24 ans respectivement, s'étaient introduits frauduleusement dans le réseau informatique de la compagnie de téléphone Bellsouth. De plus, l'un d'eux avait volé un fichier décrivant le traitement des appels au 911 (la police-secours) et les interventions de la police, puis avait téléchargé ce fichier vers un serveur RTC de Chicago, où il a sans doute fait le bonheur de bon nombre de braqueurs locaux (la ville est réputée pour sa criminalité). Tous ces appels et ces connexions longues distances étaient payés à l'aide de numéros de cartes de crédit téléphoniques détournés par le groupe. Cette condamnation marque la volonté de la justice US de freiner la délinquance informatique. Fini la rigolade pour les "hackers", dirait-on.

LE 68040 DÉBARQUE ENFIN

Après plusieurs mois d'attente, le microprocesseur 32 bits 68040 de Motorola (doté d'une unité à virgule flottante et d'antémémoires intégrées, et crédité de 15 à 20 MIPS) commence à être livré en très petites quantités (1000 par semaine). Au rythme actuel, il faudrait plusieurs décennies pour livrer les 300.000 processeurs "040" déjà commandés ! C'est pourquoi l'on espère une montée rapide de la production, malgré des problèmes de rendement. Motorola espère en produire 5000 par semaine à la fin de 1991. La principale victime de ces retards est Hewlett Packard-Appolo, dont la nouvelle gamme de machines à base de 68040 a finalement été convertie au 68030 (largement disponible par contre, mais 3 à 5 fois plus lent) avec possibilité de mise à niveau par changement de processeur dès disponibilité du 040. Autres clients notables, NeXT, qui ne fait que débuter la production de ses machines à base de 040, et Apple, dont les labos mettent au point un Mac haut de gamme basé sur ce processeur, ainsi d'ailleurs qu'un moniteur géant si lourd qu'il est surnommé "Megadeath".

À TABLE !

Pour faire bonne chère sans grever son budget, pour garder la ligne en suivant une alimentation équilibrée, Hexagone Production s'apprête à commercialiser "La Cuisine", un gestionnaire / éditeur aux mille et une recettes, doublé d'un module diététique des plus affriolants. Il était à prévoir que l'ordinateur en vienne à investir l'intimité de nos cuisines : c'est maintenant chose bien faite, au vu de l'ergonomie particulièrement réussie de ce logiciel gastronome. Consultation, gestion et édition de recettes sont manipulables à loisir sous une base de données multi-critère fort conviviale, en liaison avec un index des ingrédients et un tableau de leurs propriétés diététiques et nutritionnelles. Vous vous délecterez de la visualisation de superbes cartes (géographiques) des vins de France, calculerez avec soin vos besoins caloriques quotidiens. Ultime délice (pour la bonne bouche), la recherche dite "du frigo" : des propositions de recettes d'après sélection des ingrédients à disposition dans le garde-manger, du nombre de convives, de la chereté du plat envisagé, et d'après indication du temps disponible pour les réaliser.

LES VISIONS D'ARTHUR C. CLARKE

L'ancien chef du développement des stations de travail RS/6000 d'IBM, Andy Heller, vient de créer une firme nommée HAL Computer, en l'honneur du roman "2001, l'odyssée de l'espace" de Clarke. Les ordinateurs HAL, qui seront au point vers 1993, seront des machines multiprocesseurs sous UNIX offrant un rapport prix/puissance très intéressant. Avec humour, Heller précise que les ordinateurs HAL seront assez puissants pour faire tourner des applications d'HAL et de reconnaissance de formes. Voilà une vision prophétique. Hélas, les autres prédictions de "2001" souffriront : la base lunaire permanente ne sera certainement pas opérationnelle en la 1re année du XXIe siècle, et la fameuse navette PanAm n'a nulle chance d'exister jamais, vu la récente faillite de ce transporteur aérien.

PHILIPS EN ROUGE CEDE SA PLACE AU GRAND BLEU

Le géant européen Philips a jeté un froid en annonçant qu'au lieu de perdre "seulement" 1,1 milliard de dollars en 1990, la firme allait probablement devoir annoncer des pertes d'environ 2,3 milliards. Les branches informatique et composants ont une bonne part de responsabilité dans ce résultat désastreux, qui, outre une valse de directeurs au conseil d'administration, va entraîner plusieurs milliers de suppressions d'emplois. Les activités de prestige et de recherche de Philips avaient déjà été sérieusement réduites. Philips vient même de se retirer partiellement du programme européen de recherche Jessi (où il participait dans le domaine des composants), mettant ainsi en danger le financement du programme. Heureusement, IBM Europe (Big Blue pour les intimes) vient d'entrer dans Jessi, levant les doutes qui pesaient sur l'avenir du programme.

IBM est en effet suffisamment actif en Europe pour prendre part à ce grand projet destiné à contrebalancer les programmes de recherche américains et japonais. Le labo IBM de Zurich, par exemple, compte plusieurs Prix Nobel.

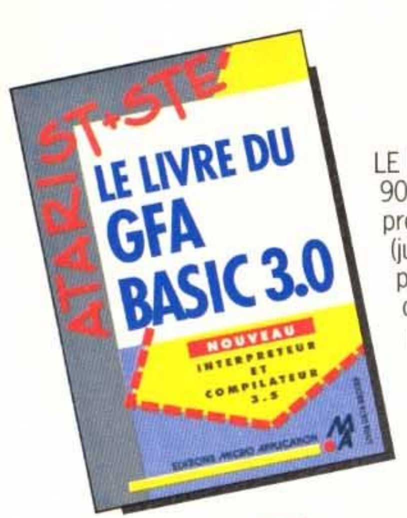
DES FINANCIERS SOUS LE SIGNE DU SOLEIL

La banque Société Générale vient de choisir les stations de travail Sun sous Unix pour équiper ses salles de marchés financiers. Autre percée pour ce constructeur, la Caisse des Dépôts et Consignations, qui gère un avoir de plusieurs milliards de francs, vient d'équiper une salle de transactions financières en matériel Sun. La bagatelle de 67 SPARCStations 1+ et de 30 SPARCservers, reliés par un réseau Ethernet, font tourner des applications financières américaines. Jadis, la seule pensée d'une machine Unix (surtout Sun qui utilise l'Unix BSD !) en gestion faisait frémir les décideurs. Et l'idée de voir un organisme d'État utiliser autre chose qu'un matériel Bull (même s'il n'était pas toujours très performant) faisait bien rire les mêmes. Sic transit gloria Bulli...

NUMERIC ART TECHNOLOGIE

Les éditions Numeric Art Technologie, vous connaissez ? Une gamme de produits qui ne manque pourtant pas d'éclectisme, jugez plutôt : création graphique, gestion de compte bancaire, finances, utilitaire de disque dur... Même les éducatifs répondent présents à l'appel : "je sais compter" et "les mots brouillés", que nous inscrivons de suite dans la catégorie des éducatifs d'éveil . En vue d'une chronique par le menu, un de nos collaborateurs émérite débrouille déjà la question.

POUR MIEUX PROGRA
DESSINER... ET MEME



LE LIVRE DU GFA BASIC 3.0
900 pages sur la programmation en GFA 3.0 (jusqu'à la version 3.5) avec plusieurs applications complètes et plus de 160 programmes et procédures détaillés. Réf. ML 571. 199 F. 928 p.
Réf. ML 671. 265 F avec la disquette.



AUTOFORMATION AUX LANGAGES BASIC
Les fonctions des langages GFA Basic, STOS et Omikron® détaillées avec 80 programmes pour vous aider à bien gérer et exploiter votre machine. Réf. ML 865.199 F avec la disquette. 360 p.



BIEN DEBUTER TEXTOMAT DATAMAT ET CALCOMAT
Apprenez à éditer vos lettres sous Textomat, à construire une gestion de fichiers sous Datamat et bénéficiez de la puissance de calcul de l'Atari grâce au tableur Calcomat. Réf. ML 784. 129 F. A PARAITRE

LE LIVRE DU REDACTEUR JUSQU'A LA VERSION 3.10
Une description complète de toutes les fonctions de ce logiciel et une partie "applications" pour vous permettre de combiner plusieurs outils et réaliser une lettre type, une facture et même un livre... Réf. ML 792. 149 F. A PARAITRE.



LE LIVRE OMIKRON® BASIC
Des modèles d'applications complètes, des utilitaires... pour réussir vos premiers pas en programmation structurée. Réf. ML 728. 165 F. 324 p.

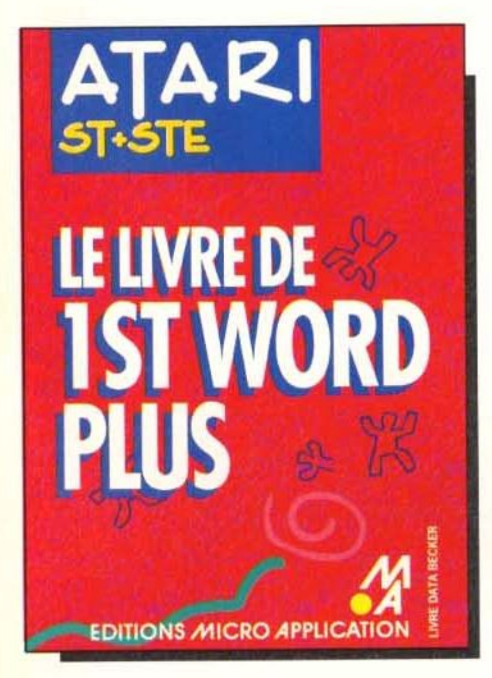
L'HISTOIRE DE LARRY
Des explications étape par étape et des solutions dévoilées en dernier recours pour suivre le héros Larry Laffer dans ses aventures mêlant humour, charme et érotisme. Réf. ML 716. 79 F. 192 p.



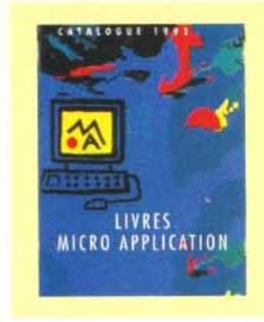
MMER,
DRAGUER.



LE LIVRE DE DELUXE PAINT Une foule de conseils pratiques pour exploiter les multiples facettes de ce formidable outil de création graphique. Réf. ML 779. 129 F. 168 p.

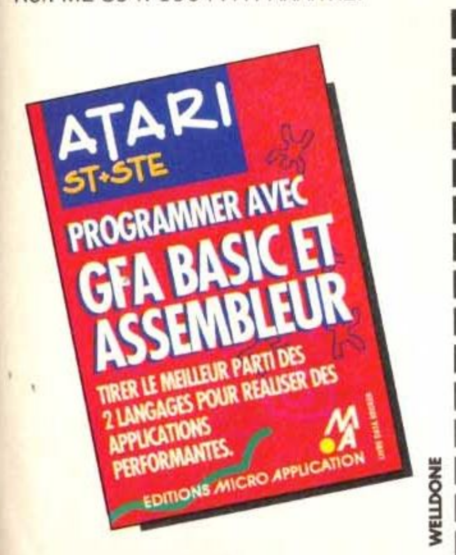


LE LIVRE DE 1ST WORD PLUS
Des informations précises accompagnées de nombreuses illustrations pour travailler efficacement sur l'un des outils d'édition les plus performants. Réf. ML 740. 165 F. 218 p.



Le meilleur moyen de rester en prise directe avec l'actualité informatique.

PROGRAMMER AVEC GFA BASIC ET ASSEMBLEUR
Toutes les informations nécessaires pour développer des applications de haut niveau avec ces deux langages.
Réf. ML 894. 195 F. A PARAITRE.



MICRO APPLICATION 58 RUE DU FG POISSONNIERE 75010 PARIS/TEL(1) 47 70 32 44

DESIGNATION	PRIX
FRAIS D'ENVOI*	
*20 F si commande inférieure à 250 F / 40 F recommandé.	
TOTAL TTC	

Nom ______

Adresse ______

Ville ______

Code postal

☐ mandat ☐ chèque
à l'ordre de MICRO APPLICATION

☐ GRATUIT :
je désire recevoir le catalogue MA 91

☐ carte bleue
date d'expiration ______

Date ______ Signature

WELLDONE

EDITIONS MICRO APPLICATION

COMME C'EST BON D'ETRE INTELLIGENT



MUSIC MASTER

LE SYNTHÉ DES HAMSTERS

Music Master vous en mettra plein les oreilles, puisqu'il transforme votre ST en synthétiseur, et aussi plein les yeux, parce qu'il est tout en couleurs. Créer un son, une mélodie, jusqu'à un morceau complet sur 16 voies ? Facile ! Vous avez un clavier MIDI ? Encore mieux ! Vous n'avez pas d'équipement audio ? Pas de problème, votre moniteur habituel suffit. Alors, on y va ?

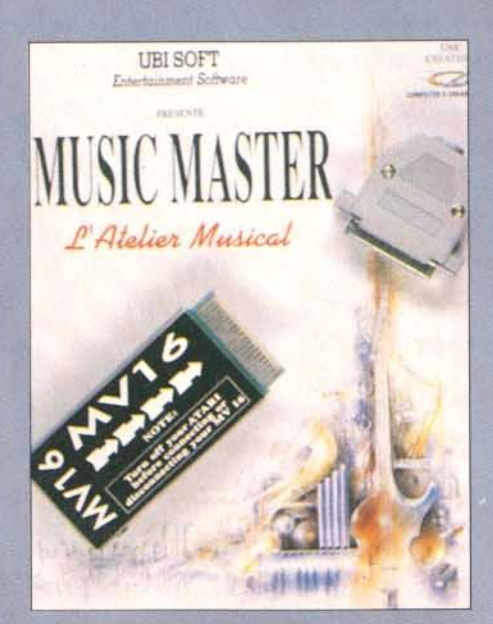
Music Master fonctionne de pair avec la carte MV16 du jeu Bat, que nous vous avions récemment présenté dans nos colonnes (cherchez bien) et dont nous avions abordé la programmation autonome dans notre numéro 38. Cette carte se connecte tout simplement sur le port cartouche du ST ; il suffit ensuite de la relier (par l'entremise d'un jack 3,5) à l'entrée audio de votre moniteur. La protection hard consiste en une clé à insérer sur le port modem, et roule !

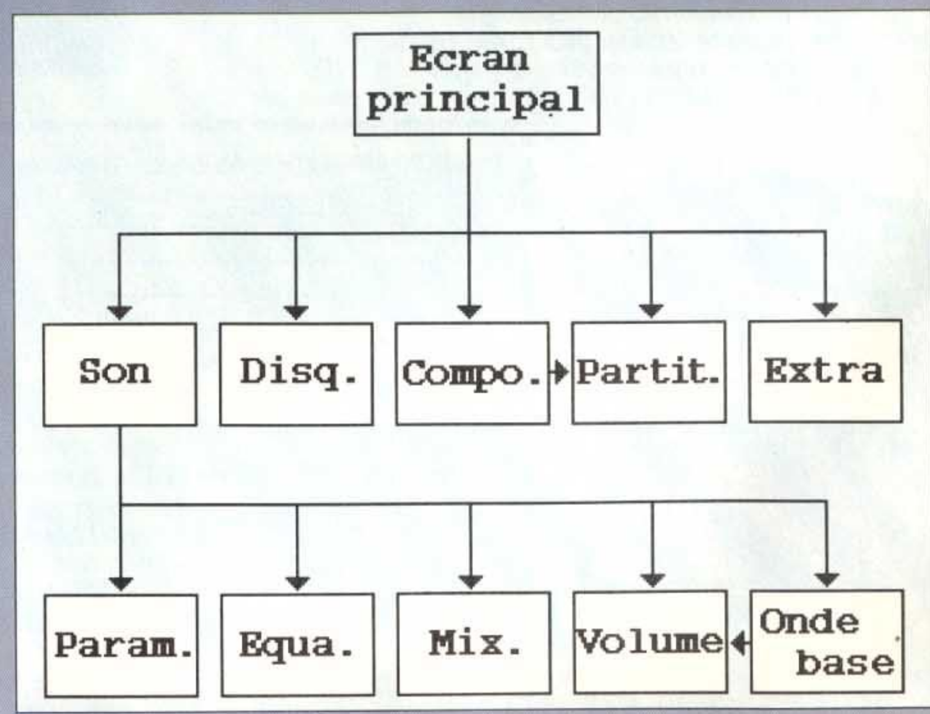
Il est accompagné de MMTools, un utilitaire qui intègre l'analyse et la récupération de sons au format IFF et ST Replay. Une disquette regorgeant d'instruments et de divers bruitages vous offre un grand choix de matière sonore pour vous faire la main sur vos premières compositions.

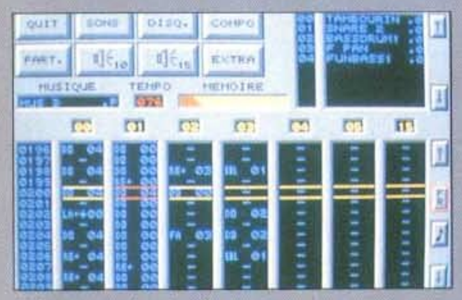
Le logiciel est entièrement écrit en assembleur ; toutes les routines sont fournies, pour ceusse qui voudraient intégrer leurs illustres créations à d'autres programmes, en ludophiles avertis.

MUSIC MASTER : VISITE GUIDEE

Attention, un écran peut en cacher un autre ! Avant que de vous perdre dans les 11 pages écran du logiciel, familiarisez-vous avec son architecture à l'aide du synoptique ci-dessous. L'écran principal donne accès à la gestion des fichiers et à l'organisation des sons entre eux, en composition directe comme au sein d'une partition. La page dédiée au son ouvre une seconde branche, consacrée à la création du son lui-même.







L'écran principal

On déboule sur cette page dès le lancement du programme. Elle permet d'une part l'accès aux différents écrans concernant l'édition d'un son, la gestion des fichiers, la composition, les partitions, et la manipulation des partitions et de la mémoire.
On peut également y créer des lignes mélodiques, dans la moitié inférieure de l'écran où s'affichent 7 pistes parmi les 16 disponibles. On inscrit une à une les notes voulues et l'échantillon correspondant à l'aide de la souris ou du clavier du ST, en se déplaçant de la même manière à l'intérieur des pistes. Une même piste peut comprendre plusieurs échantillons différents. On peut aussi choisir entre une notation française (do) ou anglosaxone (C) ; à l'extrême gauche est indiqué le temps.
A tout moment, on bénéficie d'un contrôle auditif, dont la pertinence variera suivant l'acuité auditive de l'auditeur. Deux modes de lecture sont proposés, selon le nombre de pistes utilisées : lorsqu'il y a plus de 10 pistes, la fréquence de lecture est divisée par 2.
Une fenêtre indique les échantillons présents en mémoire, en précisant leur octave. On dispose également d'informations sur le nom de la partition en cours, le tempo, et la place mémoire utilisée.



ACCES AU DISQUE

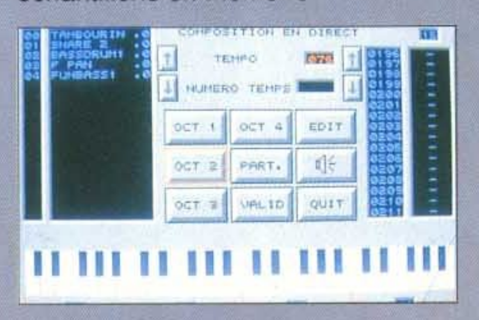
C'est ici que se gèrent les fichiers son et partition. La fenêtre du milieu vous indique les sons présents en mémoire, tandis que les deux autres énumèrent les partitions et échantillons disponibles sur la disquette. Vous pouvez choisir d'importer ou de sauvegarder une partition seule, ou avec les sons qui s'y rapportent... Vous pouvez également importer un seul ou la totalité des échantillons du disque. A l'inverse, il en est de même pour les sons créés sur Music Master, puisqu'il est possible d'en sauvegarder un seul, ou l'intégralité. Une option permet de sauver uniquement sur disquette les sons ne s'y trouvant pas déjà.
D'autres fonctions effacent ou renomment les fichiers son et partition, ou formatent encore une disquette, pour ceux qui, suite à un fâcheux oubli, se retrouveraient confrontés à des problèmes d'espace disque.
Vous pouvez travailler sur plusieurs lecteurs de disquettes, la sélection s'effectuant dans les 2 rangées alphabétiques du bas (de A à P, ça vous laisse de la marge !).
Une option de compactage est enfin prévue, pour réduire la taille de vos fichiers sur disquette lors de la sauvegarde. Attention, les sons ne pourront être relus dans ce cas-là que par Music Master.



TRAITEMENT DES SONS

Cet écran est dédié au vibrato et à la réverbération. Le vibrato est déterminé par un déplacement vertical de la souris, après l'enfoncement de la touche "Exec". On dispose de 6 types de réverbération préprogrammés comme base de départ. On peut bien sûr paramétrer le retard, la durée, ainsi que la proportion de la réverbération par rapport au son original, à l'aide des 4 potentiomètres horizontaux. Si vous êtes en manque d'inspiration, une fonction aléatoire vous soumettra diverses propositions. Dès que vous êtes satisfaits, le nouveau son peut être sauvé en mémoire sous un autre nom.

Les 5 boutons du bas donnent chacun accès à un écran spécifique pour le paramétrage du filtre, de l'équalisation, du mixage, du volume et de la forme d'onde. On retrouve dans la fenêtre de droite la liste des échantillons en mémoire.



COMPOSITION

Certains trouveront fastidieux de rentrer les notes une par une à la souris, et préfèreront composer leurs mélodies sur le clavier du ST ou, plus confortable encore, sur un clavier MIDI. Petit aparté de branchement : le Midi OUT du synthétiseur doit être connecté au Midi IN du ST, ce à l'aide d'un câble Midi normalisé.
En bas de la page figure un clavier de synthétiseur découpé en 4 octaves ; il est possible d'attribuer un son différent à chacune d'elles, en modifiant le numéro du son dans la petite fenêtre correspondante du dessous. On choisit tout d'abord un octave, qui pourra être modifiée en cours de jeu (touches 1 à 4). Les notes de la gamme sont reproduites sur le clavier du ST. Par exemple, l'octave du bas s'étend de la touche "<" à "n", les dièses ayant élu domicile sur la rangée supérieure, de "q" à "h". L'enfoncement de "Edit" vous donne un métronome de repère : l'enregistrement démarrera dès l'enfoncement d'une touche. Une fois la séquence jouée, n'oubliez pas de la valider. Les notes sont automatiquement stockées sur la piste 15, que vous pouvez réécouter. Un accès direct à l'écran Partition est prévu pour réarranger dans la foulée ce que vous venez d'enregistrer.

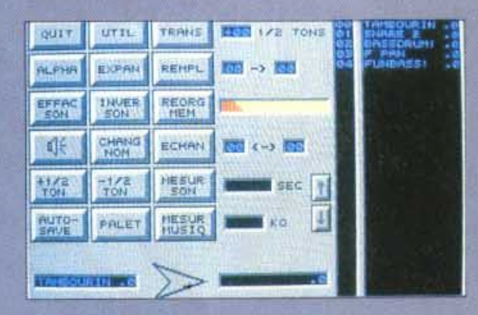
PARTITION

On retrouve l'affichage des pistes de l'écran principal. Dans le quart supérieur droit, on détermine l'état des 16 voies, "on" ou "off". De nombreux outils permettent de manipuler des





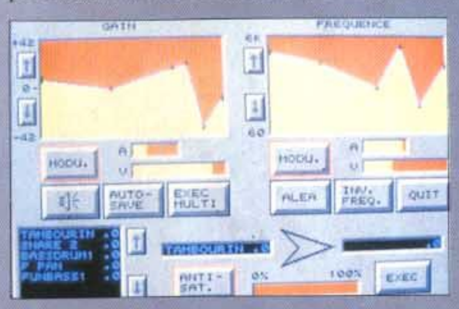
blocs : copier, insérer, effacer, supprimer. D'autres outils sont actifs sur les pistes elles-mêmes : copier, effacer.
Après avoir sélectionné l'outil, on détermine les bornes du bloc à modifier, sa source et sa destination, et le numéro de ou des pistes concernées. Pour la copie et l'insert, on peut aussi paramétrer le nombre de répétitions si nécessaire. En cas d'erreur, la fonction "undo" annule la dernière opération (en fonction de la place mémoire encore disponible).



EXTRA

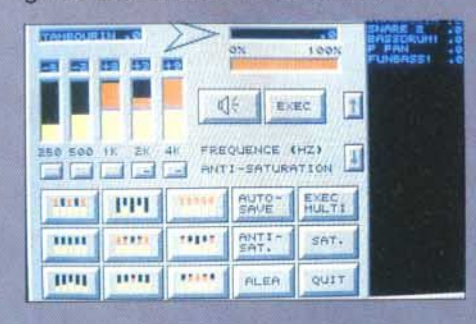
Cette page vous offre toute une série de fonctions très pratiques.
Il est possible d'y paramétrer la transposition d'un échantillon, ou d'y transposer l'ensemble de la partition par demi-tons. On peut aussi remplacer un son par un autre ou échanger deux sons, l'effacer, le renommer, ou le lire à l'envers. La liste des sons peut être classée par ordre alphabétique.
Quant à La fonction "Expand", elle "intercale un silence entre chaque son de la partition et double le tempo" (fin de citation).
Deux options sont prévues pour la sauvegarde : "Util" conserve uniquement en mémoire les sons utilisés dans la partition, et un mode "Auto" sauve systématiquement sur disquette tout nouveau son créé. Vous pouvez ensuite optimiser la mémoire en la réorganisant.
Des informations sur la durée du son (en secondes) et sur la taille de la partition (en Ko) sont disponibles.

Elles s'avèrent fort utiles lorsque vous désirez l'intégrer dans un autre programme, dans une place mémoire définie. Enfin, petit gadget, vous pouvez choisir parmi 3 palettes pour définir les couleurs de l'écran.



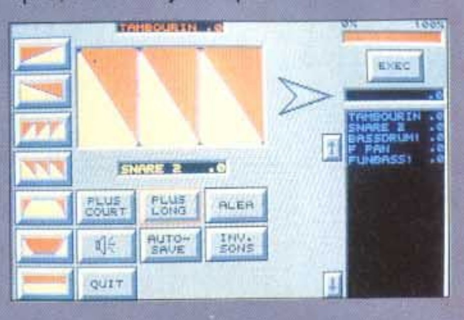
PARAMETRES

Sans entrer dans la technique à outrance, il faut simplement savoir qu'il s'agit d'un filtre passe-bande, dont on peut paramétrer l'amplitude, et la fréquence de la bande centrale ; on modifie l'évolution de sa courbe dans le temps à l'aide de la souris. En cliquant sur l'un des points, on repositionne les 2 segments qui s'y rattachent. On peut également appliquer à ces courbes une modulation, dont l'amplitude et la vitesse sont réglables. Une option permet d'inverser la courbe de fréquence.
La combinaison de tous ces paramètres offre de nombreuses possibilités de création à partir d'un même échantillon. Ici aussi, une fonction aléatoire vous permet de modifier rapidement et radicalement un son, en y incluant l'option d'anti-saturation. La transformation du son peut aussi être validée sur l'ensemble des 4 octaves, grâce à la fonction "Exec Multi".



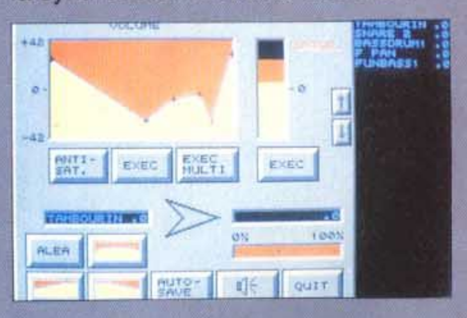
EQUALISATION

L'équaliseur permet d'amplifier ou de diminuer une "tranche" de fréquence du son. Il propose 5 bandes, de 250 Hz à 4 KHz, entièrement réglables à la souris. Neuf presets d'équalisation sont disponibles, ainsi qu'une fonction d'aléatoire. On peut valider l'anti-saturation indépendement pour chaque bande. Les deux modes d'exécution, simple et multiple, sont toujours présents.



MIXAGE DE 2 ÉCHANTILLONS

Encore plus drôle, créer un nouveau son avec deux échantillons. Il suffit de choisir un premier son dans la liste avec le clic gauche, et un second avec le clic droit. Dans la fenêtre graphique, on détermine à la souris la façon dont les 2 sons vont se mélanger. Sept presets de courbes sont fournis, ainsi que la traditionnelle fonction aléatoire.
Vous pouvez déterminer si la longueur du son résultant sera celle de l'échantillon le plus long ou le plus court. Une option permet de permuter l'emplacement des 2 échantillons sélectionnés. On arrive à des résultats très intéressants avec 2 sons de nature différentes, et à des effets subtils avec des échantillons proches ou de même famille. En répétant l'opération sur des formes d'onde simples, on fait carrément de la synthèse sonore dite "additive".

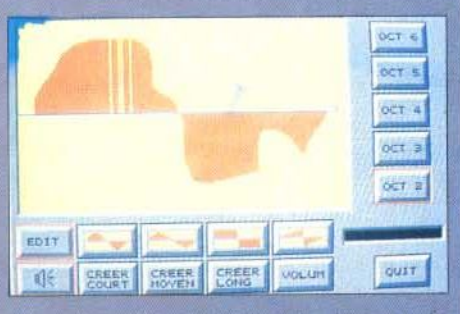


VOLUME

Il s'agit de dessiner ici la courbe du volume, dans la fenêtre graphique désormais familière. Le niveau de saturation se règle dans la petite fenêtre adjacente, et dispose de sa propre fonction d'exécution. Cela permet de retravailler un son sans toucher à son taux de saturation,

dans le cas d'une guitare saturax (vulgaris guitarus), par exemple. L'ensemble de la courbe peut se voir affecté au besoin d'une anti-saturation. Trois courbes préprogrammées sont proposées.

FORME D'ONDE

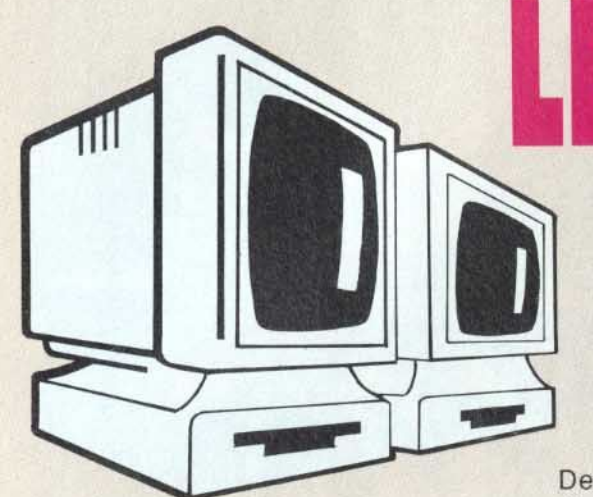


Dans cet écran, on dessine à la souris la forme d'onde d'un son. Une première écoute des 4 formes d'onde les plus courantes (sinus, triangle, carré, dent de scie) vous familiarisera avec la notion du timbre. La largeur de la fenêtre respecte l'échelle de temps selon la fréquence choisie : elle diminue donc de moitié à chaque passage à l'octave supérieure. On détermine tout d'abord quel sera l'octave de l'onde parmi les 5 disponibles. Lorsque "Edit" est sélectionné, la flèche du curseur est alors remplacée par une barre verticale. En promenant la souris dans la fenêtre tout en maintenant le clic gauche enfoncé, on entend la forme d'onde au fur et à mesure de son tracé. Quand le résultat est satisfaisant, on peut le sauvegarder sous forme de son brut court, moyen ou long (1, 2 ou 4 secondes). Un accès direct à l'écran de volume (photo précédente) est disponible.
Ces formes d'ondes simples peuvent ensuite être reprises dans l'écran de mixage, puis réadditionnées les unes aux autres pour créer un son plus riche. On peut aussi leur faire subir toute sorte de traitements de filtrage, d'équalisation ou autres, dans les pages réservées à cet effet. Tout est permis !

Music Master est idéal pour l'apprentissage du traitement du son. Il offre en effet une visualisation physique des transformations, doublée d'un constant contrôle auditif. La qualité du son est très propre, même sur un moniteur ordinaire et les performances de la carte MV16 ne sont plus à démontrer. La prise en main du logiciel reste assez simple, même si son exploration totale vous demandera un peu plus de temps. De la création d'un son à la composition finale, vous pouvez créer en vous faisant plaisir, sans être forcément ingénieur du son, pianiste virtuose, compositeur maudit, ou programmeur des étoiles (J'en oublie ?...). Le tout est de ne pas se laisser impressionner par la multiplicité des écrans et des fonctions proposées. La notice est assez bien faite, et des informations plus techniques y sont aussi disponibles pour ceux qui désireraient approfondir le sujet.

Music Master respecte donc son appellation "d'atelier de création musicale" : très pratique pour les musiques de jeu ou de démo, il répond aussi bien à des élans plus purement musicaux. Destiné au grand public, ce produit est d'un bon rapport qualité/prix, puisqu'il est annoncé pour début février à moins de 400 Frs. Music Master existe également en version PC et Amiga, avec quelques différences propres à l'ergonomie de chaque machine. Détail intéressant, les fichiers sont compatibles entre les trois versions.

Anne Olivelli



LES "TOWERS" :

Deuxième partie

Dans la digne continuité de notre série sur les recarrossages de ST, nous recevons ce mois-ci dans nos colonnes le produit de la société SCAP. En effet, avec un an d'existence, son offre est à placer dans le peloton de tête des Towers, ce qui méritait d'être signalé !

PHILOSOPHIE DE LA MÉGA TOWER

Cette réalisation a débuté assez originalement, puisqu'elle était au départ destinée à montrer à la firme Atari ce qu'attendait le public. Bien plus qu'une prouesse technique, elle visait à préfigurer ce que devraient être les futures évolutions de la gamme ST, et particulièrement le TT. Son rôle était bien sûr de démontrer l'importance du boîtier, mais aussi des extensions qu'il contenait. Ainsi, elle intégrait un des premiers lecteurs haute densité (1.44 Méga), et prouvait qu'il n'engendrait aucune incompatibilité, en acceptant toutes les protections des jeux.

Devant l'accueil qu'elle reçut sur divers salons, et les désirs des utilisateurs, SCAP décida d'en proposer une réalisation "sur mesure".

DESCRIPTION

Le boîtier utilisé est réellement une tour puisque, comme vous pouvez le voir sur sa photographie, il est à la verticale. Il peut recevoir toutes les cartes de Méga ST, mais pas celles du TT, de dimensions trop importantes. Son alimentation est d'une puissance largement suffisante (230 W), et a pour incontestables avantages de présenter l'interrupteur de mise sous tension sur la façade avant (enfin...), ainsi que d'offrir une sortie 220 V pour commander un moniteur avec ce même bouton. Sur les premières versions de cette Méga Tower, un deuxième interrupteur permettait de n'alimenter le ST qu'après l'initialisation du disque dur. La carte DMA/SCSI d'ICD étant aujourd'hui utilisée, la procédure de boot est entièrement automatisée.

La façade comprend de plus le reset, la sortie clavier, et tous les emplacements pour lecteurs de disquettes et disques amovibles. Vous trouverez ainsi trois emplacements pleine-hauteur pour des disques 5"1/4, deux 3"1/2 demi-hauteur, et enfin un dernier 5"1/4 interne. Libre à vous d'y installer ce que vous désirez, comme un lecteur haute densité ou un disque Syquest. Enfin, pour protéger vos chères mémoires de masses, un couvercle intégré au boîtier se referme sur la partie supérieure de la façade.

Après six rapides coups de tournevis sur les six vis qui referment la carrosserie, le coeur de la Tower vous est offert. Pas moins de deux ventilateurs y sont présents, l'un pour l'alimentation et l'autre pour la carte mère. La place libre disponible est importante, et les cartes grand-écran, émulateurs ou autres seront les bienvenues. Cette configuration verticale de boîtier permet d'accéder très simplement aux deux faces de la carte ST, ce qui simplifie grandement les manipulations et maintenances.

Toutes les configurations peuvent être réalisées par SCAP, leurs prix dépendant évidemment des extensions demandées. Pour tarif de base, le boîtier nu avoisine les 2200 F et atteint 4200 F avec la connectique. Ajoutez moins de 3000 F et vous aurez le plaisir de travailler avec un disque dur amovible de 44 Mégas...

Sébastien Mougey

BUDGET PERSONNEL
3 nouveaux outils

Vous êtes informaticien à vos heures perdues, vous êtes père de famille (nombreuse ?) et vous souhaitez enfin pouvoir utiliser votre ordinateur à des fins utiles. Suivez alors avec nous cette pérégrination au pays des logiciels de gestion familiale. Vous allez avoir le plaisir de faire connaissance avec Budget Famille, Budget Familial (ne pas confondre) et Compte-Chèque. Que la visite commence...

BUDGET FAMILLE

Un nouveau programme de gestion de budget familial (les fêtes sont assez loin pour parler budget) nous arrive de la douce Bretagne. Micro C, société rennaise de logiciels informatiques le commercialise pour ST sous l'appelation "Budget Famille version 2".

Aucune indication n'est donnée quant aux types de machines pouvant utiliser ce logiciel, mais il est très probable qu'il fonctionne indifféremment sur l'ensemble de la gamme ST, étant donnée sa petite taille (100 ko). Bien que cela ne figure pas non plus dans la documentation, Budget Famille travaille avec un moniteur monochrome comme en couleur : nous l'avons vérifié.

Après la mise en route, le logiciel affiche sa petite page de présentation, puis le menu qui n'est pas un menu GEM classique mais une liste d'options au milieu de l'écran. En couleur, chaque option contient une lettre d'une couleur différente, ce qui permet aux petits malins de deviner que Budget Famille dispose de raccourcis-clavier permettant par exemple de sélectionner Comptes en enfonçant tout simplement la touche C. Mais, petit oubli de programmation, il est impossible de deviner cette teinte pourtant bien utile en mode monochrome, car le menu y est écrit intégralement en noir, sans aucune différence selon les lettres. Les options, cependant, peuvent bien sûr être sélectionnées à la souris.

Le fonctionnement de ce genre de logiciel implique que l'on commence par indiquer les différents comptes bancaires que l'on a à sa disposition, Budget Famille permet d'en gérer 100 individuellement, ce qui personnellement nous a suffi. Tous les traitements se font à partir de trois critères : les comptes bancaires, les budgets et les affectations, aussi appelés libellés. Pour chacun de ces critères, cent choix sont programmables, ce qui donne des possibilités quasi-illimitées (nous considérons toujours le père de famille tranquille et non pas un Marcos ou autre dictateur richissime).

Mais attention : le mode d'emploi, pour lequel l'imprimeur semble avoir oscillé entre le format A5 et A4, se résume à une malheureuse page recto verso. Les explications qui y sont données sont assez incompréhensibles, un peu à l'image de l'utilisation du logiciel. Il n'est par exemple pas évident de faire la différence entre la saisie par relevé et la saisie par opération, et ce n'est le "manuel" qui vous y aidera.

Faut-il l'imputer à la programmation ? La manipulation de Budget Famille

Budget Famille 2.00			Feuille de relevé		Date du relevé : 12/12/90	
Compte : CL			Ancien solde :		13542.00	
Opération	Numéro	Date	Budget	Libellé	Débit	Crédit
REMISE CHEQ.		171290	BOURSE			5000.00
RET			DIVERS		250.00	
CHE			EAUX	FNAC	2500.00	
FIN	Opération en cours : 5/ 3			Solde :		15792.00 F

Opérations
- Chèque
- Carte de crédit
- Retrait
- Prélèvement
- Virement banque à banque
- Frais bancaires
- Remise de chèques
- Virement
- Versement



est en tout cas fort malaisée : on a toujours l'impression de cliquer sans que les effets désirés s'ensuivent. La logiciel ne brille pas non plus par sa puissance : il se contente de faire des additions (plus quelques soustractions, ne soyons pas mauvaise langue) ; la plupart des autres logiciels de ce type permettent au moins de calculer les pourcentages des dépenses selon les affectations, mois par mois et graphique à l'appui. Ici, foin de pourcentage ou de graphiques. Un produit qui rappelle le type de logiciel utilisé par les banques pour faire leurs relevés.

Si donc votre but est d'avoir votre relevé de compte avant même que la banque ne vous l'envoie, courez acheter le logiciel, sinon, attendez que la banque fasse son métier.

Budget Famille ne fonctionne que sur disquette, en sauvant ses données sur une deuxième disquette, ce qui ne manque pas d'arriver à chaque fois que l'on entre quelque chose. Cette pratique d'accès systématique vers l'extérieur permet une très grande fiabilité en cas de panne de courant (mais ce n'est guère courant...), seules les toutes dernières données sont perdues, mais de ce fait, l'ordinateur passe plus de temps à faire des accès disque qu'à être disponible.

En regardant la disquette de sauvegarde, on s'aperçoit que le logiciel sauve ses données sous la forme d'une multitude de tout petits fichiers, contenant l'un le solde d'un compte, l'autre le contenu d'un relevé, etc. La taille des fichiers est ainsi bien souvent de moins de 100 octets ; lors de l'essai, des fichiers de 0 octets furent même créés, on se demande bien pourquoi. Sans doute une astuce de programmation bien cachée, en attendant qu'une future version de Budget Famille version en tire parti.

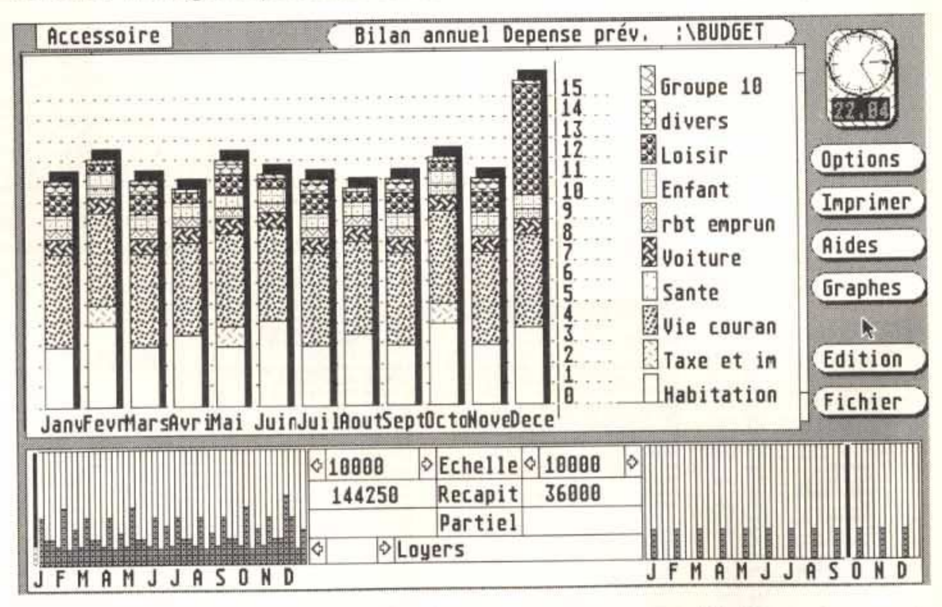
Dernière chose : de retour sous le bureau, en couleur, le fond est tout à coup devenu bleu, et en mode monochrome, passé en inverse vidéo. Avis aux programmeurs : lorsque l'on décide de changer la palette du ST, la moindre des choses est de la reconfigurer avant de rendre la main au TOS. Amis banquiers, à vos bourses, Budget Familie vous permettra de travailler comme au bureau, avec une puissance digne du Néolithique Informatique et une ergonomie à faire pâlir le serveur de la Caisse d'Epargne. Admirez enfin la sobre élégance de la feuille de relevé de la figure 1, c'est la plus belle du logiciel...

BUDGET FAMILIAL

Poursuivons sans dévier d'un pouce notre investigation budgétaire avec Budget Familial. Peu de sociétés parisiennes semblent se battre sur ce créneau, nos amis de la France profonde (sans aucune médisance) en sont les spécialistes, puisque Budget Familial nous arrive de Cagnes s/Mer. Les veinards doivent bien bénéficier de 365 jours de soleil par an (au moins !).
Revenons à nos moutons : comme le veut la tradition, Budget Familial fonctionne lui aussi sur l'ensemble de la gamme ST, STE, Mega, et consorts ; le logiciel peut être utilisé en couleur ou monochrome. Seule ombre au tableau, nous avons aussi tenté de le lancer sur TT (notre vice est sans limites...), mais deux petites bombes sont venues interrompre notre béatitude céleste, juste après le chargement du logiciel.
Il n'en est pas moins vrai que Budget Familial peut aussi s'installer sur un disque dur et y sauvegarder ses données. Le programme étant assez bien écrit, il suffit de mettre ses fichiers dans le dossier de son choix et au lancement, Budget Familial retrouvera en un tournemain tout ce dont il a besoin. Les chemins d'accès sont même directement mis en mémoire pour tout chargement ultérieur de données.

GEM À CONSONNANCE GfA

Comme la plupart de ses petits camarades, Budget Familial n'utilise pas les menus GEM classiques, la page principale contenant des options en bas à droite de l'écran (figure 2). On aurait certainement préféré des menus normaux mais ceux-ci posent des problèmes aux programmeurs en GfA, leur gestion n'étant pas évidente. En conséquence de quoi, il faudra cliquer sur les boîtes dessinées en GfA qui ont la désagréable habitude de ne pas toujours passer en inverse vidéo au bon moment, ce qui donne l'impression de devoir cliquer à côté de la boîte pour la sélectionner.
Un bon point tout de même pour la présence de l'item Accessoire qui est le seul du menu GEM, et qui assure, comme son nom l'indique, l'accessibilité aux accessoires. Aucun problème de conflit n'est apparu entre les accessoires et le programme principal, même avec Multidesk ou AT Speed, spécialistes patentés de la chose. De tout petits problèmes de chevauchement de fenêtres se posent de temps à autre : la finition n'est pas parfaite. Ainsi, l'auteur de Budget Familial n'a pas jugé utile de configurer la taille du menu acces-



soire en fonction de la longueur du nom de l'accessoire chargé. "G+PLUS Accessory" sort allégrement du cadre, les deux dernières lettres (le r et le y pour ceux qui suivent) restant au milieu de l'écran lorsque le menu est refermé. Ce n'est pas très joli...

UN BUDGET COMPLET

Après avoir, comme précédemment, configuré nos différents comptes en banque (jusqu'à dix), dont l'un doit être dénommé espèces, le programme fonctionne en deux temps : tout d'abord, il est nécessaire de faire un budget prévisionnel sur l'année, en ventilant ses dépenses dans les différentes rubriques que l'on crée. Cette énorme prise en compte de paramètres est peut-être la plus difficile, mais aussi la plus puissante du programme. Chaque recette et chaque dépense doit ainsi être répertoriée pour tirer parti des possibilités de Budget Familial.

Une fois cette délicate opération effectuée (imaginez simplement tous les achats que vous faites pendant un mois...), il faut comme de bien entendu utiliser le logiciel à chaque nouvel achat, retrait d'argent ou entrée de fonds. A ce moment-là, le programme vérifiera si vos dépenses sont compatibles avec le budget prévisionnel type qui a été formé au début.

Deux possibilités existent : soit tout va bien, votre budget correspond exactement à vos prévisions, dans ce cas, vous n'avez plus rien à faire avec Budget Familial, et vous pouvez le jeter puisque vous êtes devenu un Dieu de la finance. Soit, deuxième possibilité, oh combien la plus probable, Budget Familial annonce fièrement que vous avez dépassé votre budget pour tel ou tel poste. Au moyen de graphiques de toute sorte (camembert, histogramme ou tableaux de données), on décèlera rapidemment ce qui ne va pas.
Non content de vous montrer vos erreurs de gestion, Budget Familial se fait fort de vous proposer des solutions ! En effet, pour chaque poste de dépense, on indique un coefficient (dit coefficient de compressibilité) permettant de calculer les économies possibles sur ce poste. Bien entendu, certaines dépenses (ou recettes) sont fixes et ne pourront pas être ratatinées, comme c'est le cas pour les loyers, l'assurance de la voiture et autres joyeusetés !

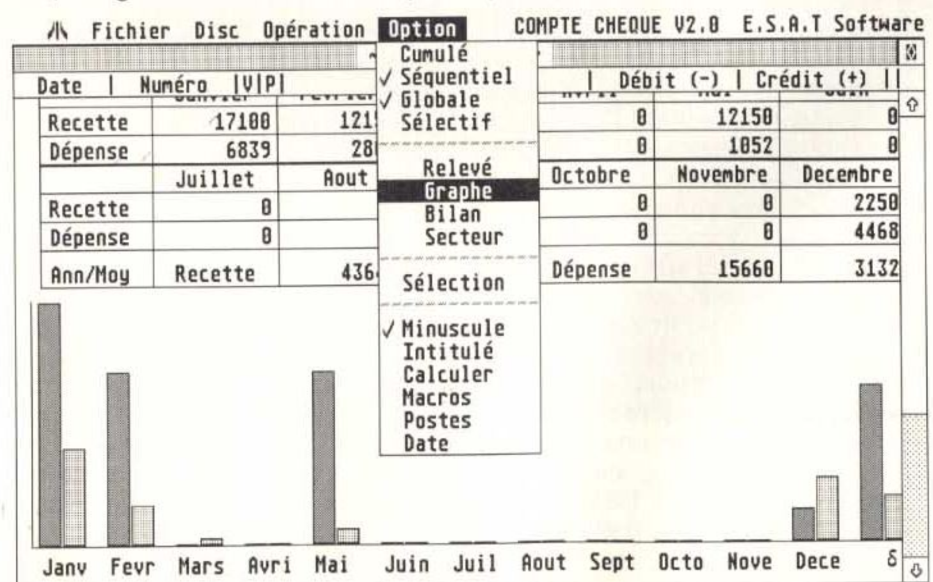
DES PLUS ET DES MOINS

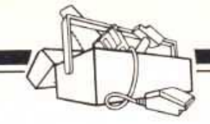
Sur la figure 3 du Budget Familial, la partie inférieure de l'écran représente le budget prévisionnel "recettes", un des points de départ du fonctionnement du budget. Au premier abord, il n'est pas très facile de s'y retrouver, mais toutes les prévisions doivent être entrées par ce graphique, en indiquant le type (recette ou dépense), la somme, la périodicité (mensuelle, hebdomadaire et bien d'autres) et le poste concerné. Le petit graphique représente les mouvements d'argent du poste concerné.

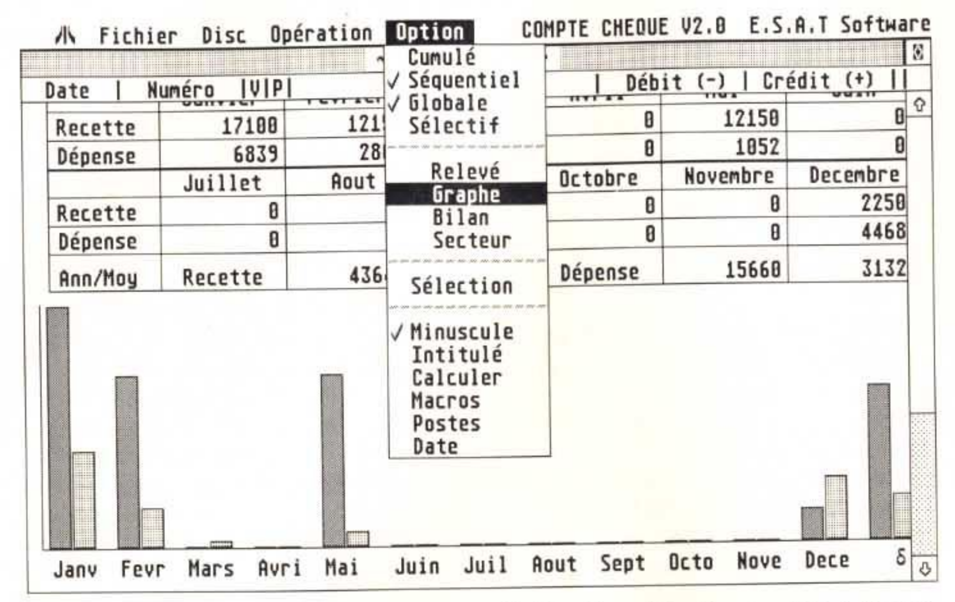
Première remarque, l'année n'est pas divisée en 52 semaines comme le voudrait la religion chrétienne mais en 12 mois de 4 semaines, ce qui fait après un rapide calcul la modeste somme de 48 semaines par an ! Il faut donc un petit peu jongler avec les périodicités des recettes et des dépenses, si l'on veut avoir un budget équilibré sur l'année; sans oublier que ce petit travail doit avoir lieu sur les deux prévisionnels (un pour les recettes, l'autre pour les dépenses), puis sur le budget réel.
D'un point de vue strictement convivial, Budget Familial ne s'en sort pas trop mal, même si la mise en page des graphiques n'est pas des plus lisibles (les noms des mois se chevauchent un peu trop à mon goût.). Extrêmement bizarrement, il n'est pas possible d'avoir accès aux accents, qu'ils soient aigus, circonflexes ou graves. C'est fâcheux.

La doc de 40 pages passe ce point totalement sous silence. En parlant de la documentation, il faut dire qu'elle n'est pas trop mal faite, avec un plan à suivre "pas à pas" pour bien comprendre les différentes possibilités du logiciel qui sont effectivement assez nombreuses. Il est néanmoins regrettable qu'aucune photo d'écran n'y figure, en particulier pour expliquer de façon plus claire les différentes fonctions des boutons des budgets prévisionnels.

La version que nous avons testée n'était malheureusement qu'une version de démonstration bridée, où toute sauvegarde était impossible, nous ne pouvons donc rien dire quant aux fichiers sauvegardés. Les exemples fournis semblent cependant indiquer qu'il s'agirait de fichiers .CPT (pour les comptes), .PRM (pour les paramètres) et .PRV (pour les budgets prévisionnels).
Une version Amiga de Budget Familial est dorénavant disponible, sans doute au même prix que la version ST, soit un peu moins de 300 francs. Voilà qui a le mérite ne pas trop grever le budget familial !







COMPTE-CHEQUE

Argh ! Voici bientôt venue la fin de de cet émoustillant point de vue sur les logiciels de gestion. "Enfin !" s'exclameront la plupart des lecteurs ; "Déjà ?" diront pour leur part les mauvaises langues. Le dernier logiciel testé a cependant reçu un joli nom de baptême : Compte Chèque. Ce programme est commercialisé par Esat Software, plus connu pour des productions comme Paint Designer, logiciel de dessin monochrome que nous vous avions présenté il y a peu.

Compte Chèque, contrairement à ses petits camarades, n'existe pas seulement sur ST, mais aussi sur PC et Amiga (cette dernière version semble d'ailleurs la plus complète). En ce qui concerne la mouture ST, toutes les configurations sont acceptées, mais des problèmes ont été rencontrés lors de l'utilisation conjointe d'un disque dur et du TOS 1.4, qui, il est vrai, n'est pas distribué officiellement en France.

Première remarque : Compte Chèque reprend le menu GEM standard, avec les différents items à leurs places "normales" (option "quitter" sise en bonne place). La page principale, que l'on peut voir en figure 4, contient une sorte de feuille de tableur avec les toujours classiques ascenseurs GEM, les différentes cases regroupant de leur côté les informations Date, Numéro, Validation (V), Poste (P), Libellé, Crédit et Débit.

Lors de la saisie d'une dépense ou recette quelconque, le programme affiche une boîte de dialogue qui permet en fait de remplir le tableau indirectement : on aurait sans doute préféré cliquer sur la case concernée, mais ce n'est hélas pas le cas. Néanmoins, cette façon de faire permet d'avoir plus facilement accès aux choix préprogrammés, puisque la boîte de dialogue donne la liste des Numéros, Libellés et Postes disponibles.

Dans Compte Chèque, les saisies de mouvement se font en deux temps : tout d'abord, on entre une dépense au moment où elle a été faite, mais dans le cas d'un chèque, la banque ne tire pas immédiatement la somme correspondante. Ici, c'est exactement la même chose, la somme est simplement enregistrée par Compte Chèque mais pas encore validée, c'est-à-dire pas encore tirée sur le compte. Après avoir reçu confirmation par la banque du retrait, vous pourrez valider la ligne en cliquant dessus et l'opération sera terminée.

Les lignes du tableau scrollent automatiquement lorsque le curseur se rapproche du haut ou du bas du tableau, il n'est donc pratiquement jamais nécessaire de jouer avec les ascenseurs. En revanche, en ce qui concerne la visualisation du tableau, le poste budgétaire n'apparaît pas clairement, seule sa lettre codée est affichée. On est obligé, pour plus de renseignements, soit de faire référence au libellé (qui donne le destinataire), soit de modifier la ligne pour simplement voir le nom complet du poste.

Les opérations de saisie, modification, effacement d'enregistrement peuvent être sélectionnées au choix dans le menu GEM, ou par un très simple raccourci clavier, bien plus agréable à utiliser.

Une fois les opérations saisies, le plus intéressant est de pouvoir les étudier plus à fond. Pour cela, différentes options sont disponibles, du tri simple à la sélection multicritère. Compte Chèque permet de confectionner un masque multicritère agissant comme un filtre sur les différents champs d'un enregistrement. Après le passage dans cette moulinette, seules les lignes correspondantes resteront affichées, elles pourront ensuite être imprimées ou visualisées graphiquement (question graphique, le résultat n'est hélas pas des plus réussis). Enfin, le bilan permet de voir le total des recettes et des dépenses par mois et année, en prenant en compte partie ou totalité des enregistrements.

Compte Chèque n'est ni très puissant, ni très joli, la documentation de 15 pages au format mini se contente de passer en revue une par une les options assez peu nombreuses des menus, sans grand souci pédagogique. Les graphiques font eux aussi pâle figure avec un histogramme maigrichon et des camemberts en une seule couleur. En revanche, Compte Chèque peut se targuer d'être le plus simple d'utilisation et donner des résultats assez honorables dans ce créneau désormais saturé des gestionnaires bancaires.

François Pagès

LES OUTILS DE DEVELOPPEMENT DE VOTRE ATARI

dBMAN V
Version 5.2 R

- Compatibilité programmes et fichiers avec dBASE.
- Puissance : plus de 360 commandes et fonctions.
- Compilateur intégré.
- Plus de 78 versions existantes.

dBMAN est le seul SGBD du marché compatible avec dBASE permettant la compatibilité de vos fichiers et applications sous environnement TOS, DOS, réseau Netbios, Novell, et Unix. La puissance de dBMAN donne à votre Atari des capacités de gestion insoupçonnées.

1.950 F HT

HISOFT DEVPAC
Version 2.20

- L'assembleur leader sur Atari.
- Version ST
- Environnement de développement avec éditeur, macro-assembleur GenST, éditeur de liens et débogueur symbolique MonSt.
- Compatible avec le Lattice C et le Hisoft Basic.

DEVPAC est l'assembleur utilisé par tous les plus grands développeurs sur Atari

875 F TTC

LATTICE C
Version 5.1

- Un leader mondial du langage C pour votre Atari.
- Version ST (version TT en cours).
- Système de développement au standard ANSI avec éditeur, assembleur, compilateur, éditeur de liens et débogueur.
- Bibliothèques très complètes : AES, VDI, BIOS, XBIOS, GEMDOS, LineA, ANSI, Lattice et UNIX.

Le Lattice C en version 5.1 vous fait bénéficier de la puissance et la notoriété d'un des "grands" du langage de développement. Les très nombreuses bibliothèques, éprouvées et appréciées par de nombreux utilisateurs sur Workstation Unix et compatible PC, faciliteront la maintenance et le portage de tous vos développements.

2.150 F TTC

LA CAO/DAO SUR ATARI

ZZ-VOLUME 1.5

- La CAO 3D volumique du bâtiment.
- ZZ-Volume allie facilité, fonctionnalité et puissance.
- Son prix est 10 fois inférieur aux produits équivalents sur d'autres micro-ordinateurs.
- La version 1.5 fonctionne sur Atari TT, ST, cartes couleurs...

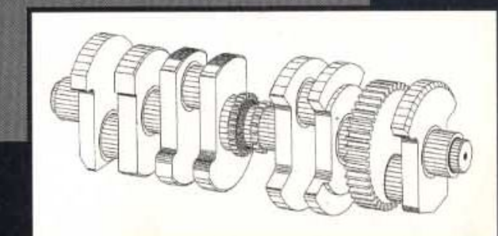
8.950 F HT



ZZ-3D

- La seule CAO 3D Volumique booléenne sur micro.
- ZZ-3D allie facilité, fonctionnalité et puissance.
- Son prix est 10 fois inférieur aux produits équivalents sur stations de travail Unix.
- ZZ-3D possède une version ST et une version spécifique TT utilisant la pleine puissance du TT..

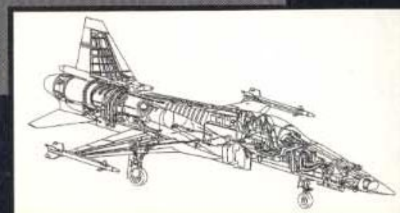
6.950 F HT pour Atari ST
8.950 F HT pour Atari TT



DynaCADD

- Le 2D générique pour toutes les professions techniques.
- L'association de DynaCADD et de l'Atari donne une table à dessin électronique d'une performance incomparable.
- Son prix est 5 fois inférieur aux produits équivalents sur d'autres micro-ordinateurs.
- DynaCADD fonctionne sur Atari TT, ST, cartes couleurs...

9.450 F HT



Packs CAO/DAO

Prix spéciaux pour les packs CAO/DAO :

- ZZ-Volume + DynaCADD : 16.450 F.HT
- ZZ-3D ST + DynaCADD : 14.950 F.HT
- ZZ-3D TT + DynaCADD : 16.450 F.HT

Les produits Human Technologies sont disponibles chez tous les bons revendeurs Atari. Pour connaître votre revendeur le plus proche, téléphonez au (1) 46 04 88 71, ou ***3615 code HUMAN.***



ONE-O-ONE 46 04 88 78

MULTIMEDIA & VIDEO DIGITALE

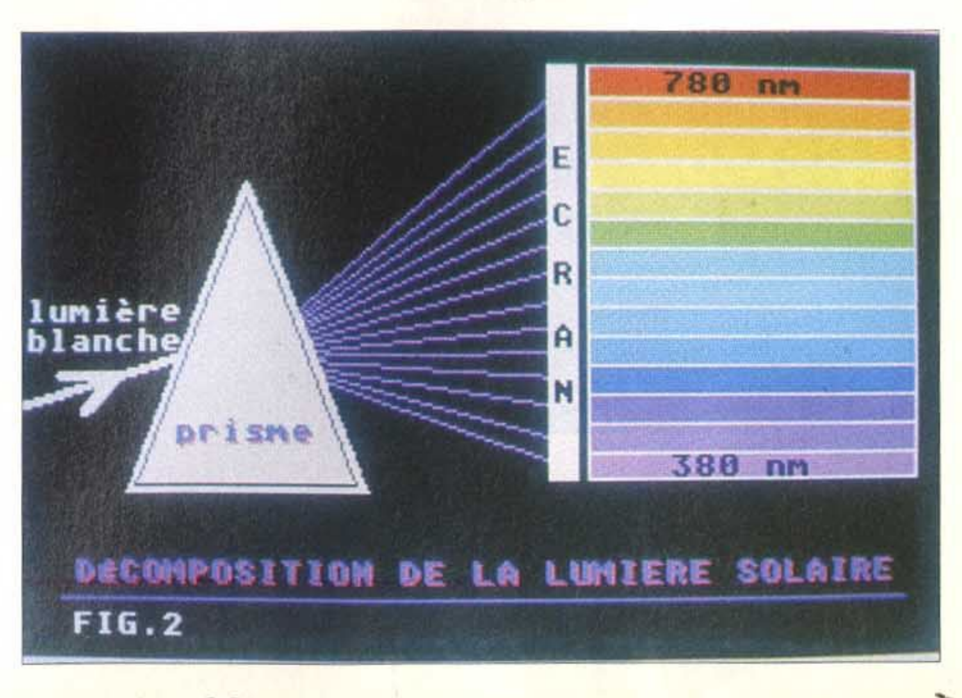
La vidéo : mot magique régnant sur un domaine en constante expansion, où les applications ne sont limitées que par des frontières technologiques sans cesse repoussées. Nous allons lever un coin de voile sur ses possibilités insoupçonnées, en vous proposant une initiation sur l'ensemble des interfaces connectables à votre ordinateur. L'arrivée récente en France de la carte Chili nous pousse aujourd'hui à vous proposer une première initiation à la vidéo, ainsi que le premier épisode de l'étude de cette carte passionnante, à suivre dès nos prochains numéros. Commençons par une petite initiation au signal vidéo avec ses caractéristiques, ses diverses composantes, et son domaine d'utilisation, pour vous permettre de mieux vous situer...

Il était une fois un signal nommé vidéo (vidéo composite) fort sympathique, qui voyageait à travers un magnétoscope pour aboutir sur l'écran de votre télévision. Ce signal, qui se déplace à votre insu d'appareil en appareil, contient toutes les informations nécessaires pour construire l'image que l'on observe sur tout écran, moniteur d'ordinateur ou téléviseur. Il comprend un tas d'informations qui vont être triées à l'arrivée (comme des lettres) pour constituer une image stable et de qualité. Comme beaucoup d'autres machines vidéo, le magnétoscope fabrique et transmet un signal dit vidéo composite ; tel un chef d'orchestre devant une partition, les moniteurs et téléviseurs se chargent, dès réception du signal, de décrypter les informations qu'il contient et de coordonner tous les mécanismes et circuits en conséquence, pour afficher sur votre écran l'image de vos rêves ; Jusque-là, rien de bien compliqué.

Mais ce signal se compose de plusieurs éléments : informations de degré de lumière (Luminance), de degré de couleur (Chrominance), et de tops techniques servant à synchroniser le tout (Synchro Horizontale et verticale)... Voilà qui commence à corser les choses !

COMPOSANTES

I) La ***Luminance*** est définie par la décomposition d'une image en une multitude de points plus ou moins gris, ordonnés selon un critère bien précis, un peu comme en imprimerie où les points sont plus ou moins serrés pour reproduire une illustration. En vidéo,



Décomposition de la lumière solaire

LES PERIPHERIQUES PROFESSIONNELS DE VOTRE ATARI

NOUVEAU

LES PREMIERS PERIPHERIQUES DE L'ATARI TT

ZZ-SCREEN MP21-TT

- Ecran 21", anti-reflet, « Paper-White ».
- Accepte la haute résolution TT en 1280 x 960 monochrome.
- Accepte les résolutions inférieures (640 x 400, 320 x480...).
- Les couleurs sont traduites en niveaux de gris réels.

11.450 F HT



ZZ-SCAN MC-3256

- Le premier scanner spécifique pour l'Atari TT.
- Format A4, mécanique CANON IX-30 F
- 300 dpi et 256 niveaux de gris réels (8 bits/pixel).
- Livré avec ZZ-LazyPaint (retouche N&B), ZZ-Erox (photocopie), ZZ-Scan (scan en niveaux de gris).

9.950 F HT



Autres modèles :

La technologie scanner couleur Epson sur Atari ST et TT.

ZZ-SCAN CE-216/A7 9.950 F HT
A7, 200 dpi, 16 millions de couleurs : scan couleur, ZZ-LazyPaint, ZZ-Scan, ZZ-Erox.

ZZ-SCAN CE-416 17.900 F HT
A4, 400 dpi, 16 millions de couleurs : scan couleur, ZZ-LazyPaint, ZZ-Scan, ZZ-Erox.

ZZ-SCAN CE-616 20.900 F HT
A4, 600 dpi, 16 millions de couleurs ; scan couleur, ZZ-LazyPaint, ZZ-Scan, ZZ-Erox.

SCANNER ZZ-SCAN MC 332

- Le leader du marché Atari à un prix bureautique.
- Format A4, mécanique CANON IX-12 F.
- 300 dpi, 32 niveaux de gris.
- Livré avec ZZ-LazyPaint et module de photocopie.
- Le scanner idéal pour la bureautique éditoriale.

8.450 F TTC
Nouveau Prix





ZZ-SWITCH

- Gagnez plus de 10.000 F sur l'achat d'une 2ème station laser Atari.
- 2 Méga ST sur une laser Atari.
- Carte électronique de gestion du DMA.
- Possibilité d'utiliser des câbles jusqu'à 3 mètres de longueur.

1.995 F HT

ECRAN HAUTE-RESOLUTION ZZ-SCREEN MP19-ST

- Ecran 19", anti-reflet, « Paper White ».
- Carte M110 Matrix, 1280 x 960, monochrome, processeur Hitachi.
- Fonctionnement en 1280 x 960 ou émulation 640 x 400 Atari (zoom).
- Connexion sur BUS Méga ST.

12.950 F HT

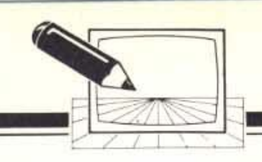
Autre modèle : ZZ-Screen MP21-ST 18.950 FHT

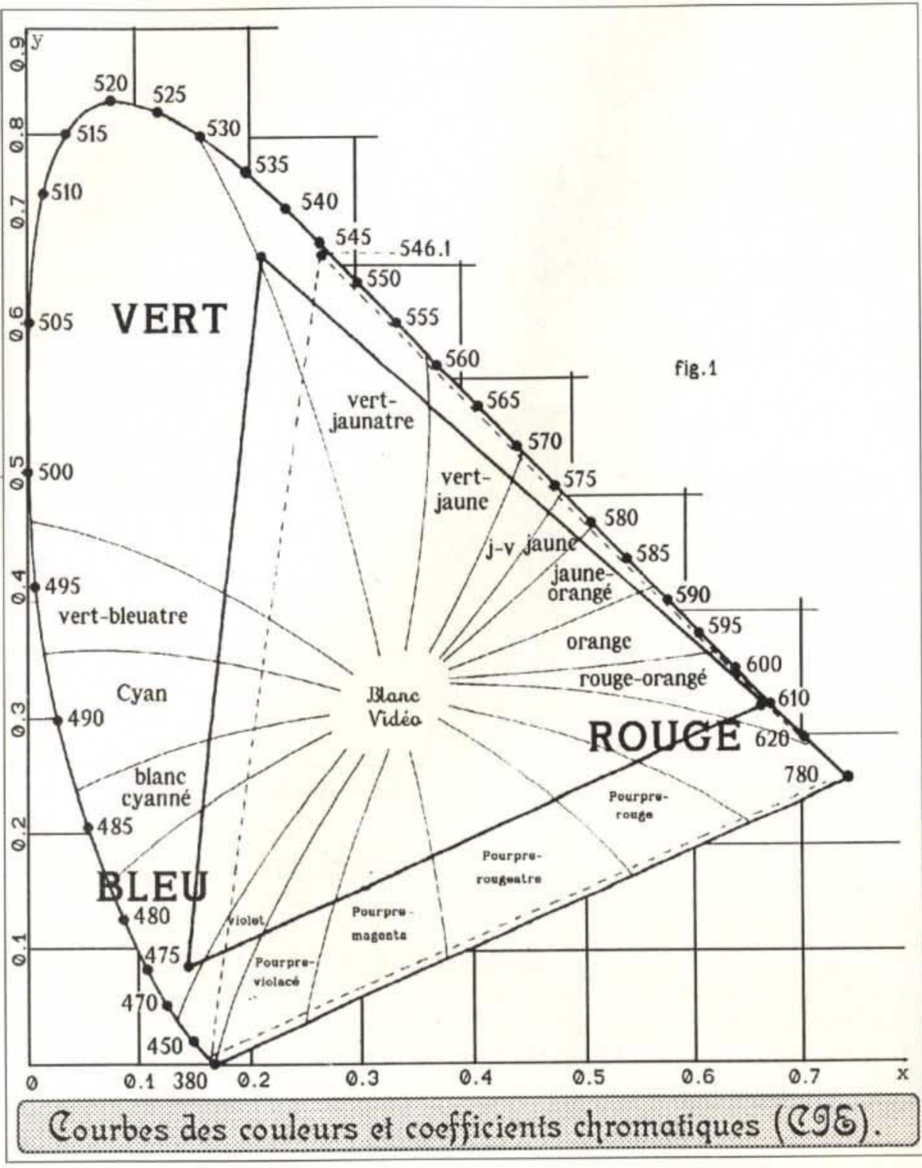
Les produits Human Technologies sont disponibles chez tous les bons revendeurs Atari. Pour connaître votre revendeur le plus proche, téléphonez au (1) 46 04 88 71 ou 3615 code HUMAN.



* Les marques citées sont déposées par leur propriétaire respectif.

ONE-O-ONE 46 04 88 78





Triangle des couleurs de la commission internationale de l'éclairage

c'est la même chose, sauf que les variations correspondent à des signaux électriques d'intensités différentes. C'est la luminance qui définit par défaut toute image noir et blanc. C'est cette même luminance qui détermine en digitalisation les niveaux de gris, suivant la qualité du convertisseur Analogique/ Digital utilisé.

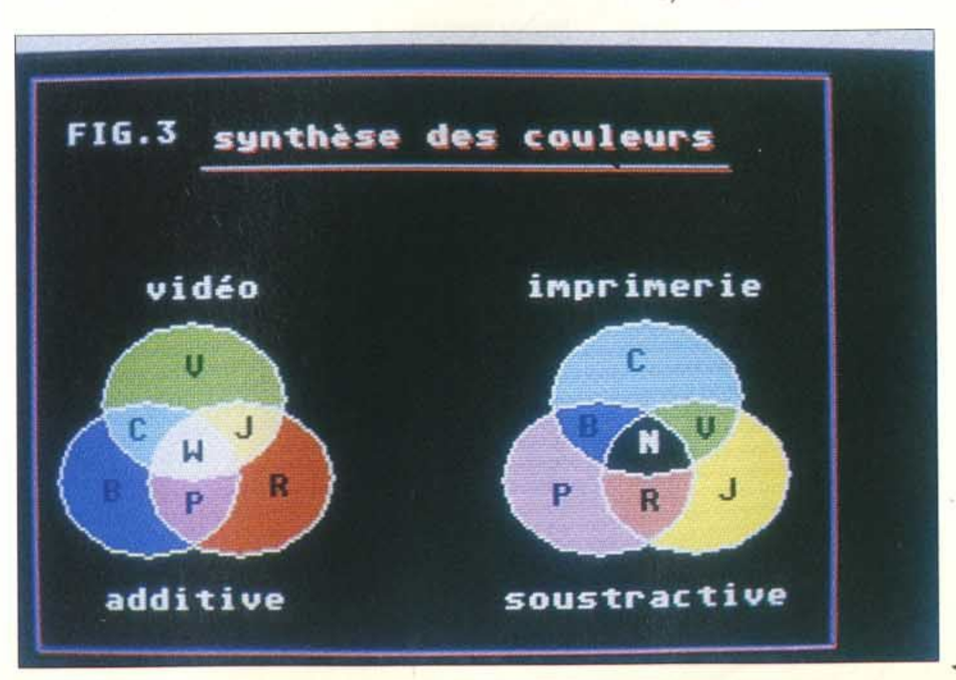
II) La ***chrominance*** contient, sous forme codée, les informations de couleur correspondant à chaque point du signal de luminance. Toutes les couleurs existantes et perceptibles par notre oeil sont obtenues en vidéo par mélange additif des trois couleurs primaires : rouge (R), vert (V) et bleu (B). A chaque point de luminance correspond une information de chrominance (voir ci-dessus).

C'est donc la notion de chrominance qui définit les différentes couleurs que nous pouvons voir dans notre environnement de tous les jours. Pour mieux comprendre ce principe, un peu de théorie est nécessaire : le spectre de la couleur est exprimé à partir d'un prisme traversé d'une lumière blanche. La projection des rayons sur un écran blanc donne une décomposition de couleurs allant du rouge perceptible par l'oeil au violet, en passant par une gamme de couleurs intermédiaires ; la même constatation est faite lorsqu'on regarde un arc en ciel. On constate cependant l'absence de certaines couleurs, comme le pourpre : cette couleur est issue d'un mélange additif des trois couleurs fondamentales (RVB). Il faut savoir aussi que chaque couleur possède une unité de mesure exprimée en nano-mètres, déterminant sa longueur d'onde. Deux exemples concrets : le Rouge vaut 780nm, et le Bleu 380nm - on en déduit que la plage du spectre visible est comprise entre 380 et 780 nm. Au-delà de ces valeurs, on rentre dans la partie du spectre des couleurs non visibles par l'oeil humain, comme l'ultraviolet et l'infrarouge.

ROUGE-VERT-BLEU, OU JAUNE-MAGENTA-CYAN ?

Voici enfin de quoi éclairer les maquettistes et infographistes côtoyant de près le domaine de

Synthèses des couleurs



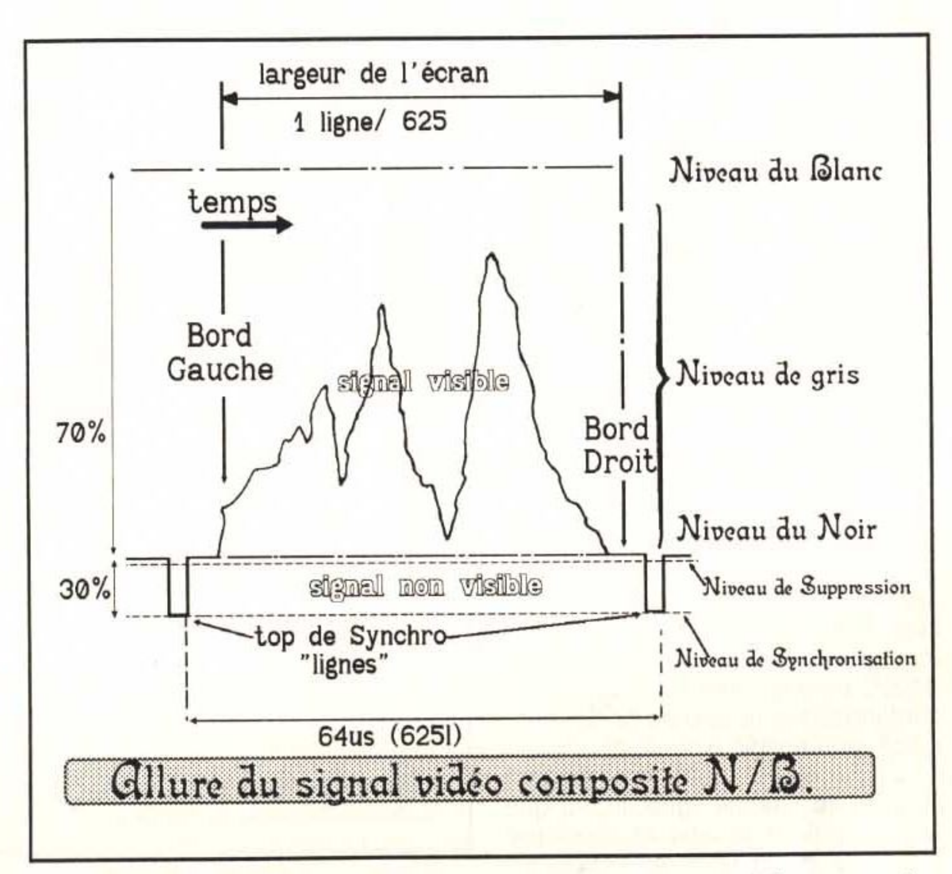


l'informatique et de l'imprimerie, qui se tirent souvent les cheveux devant la transcription des couleurs de l'écran vers le support papier. Il faut avant tout savoir que la colorimétrie utilisée sur le papier (dessin, magazines, prospectus, imprimerie générale...) et celle que nos yeux perçoivent, sont interprétées de la même manière, mais conçues différemment : dans le premier cas, il s'agit de synthèse soustractive et dans le second, de synthèse additive.

La Synthèse additive : si on utilise deux projecteurs ayant pour couleurs respectives le Rouge et le Vert, on obtiendra comme résultat une lumière Jaune ; de même, une lumière Bleue et Verte donnera un éclairage de couleur Cyan. On constate en effet que le Jaune est situé entre le Rouge et le Vert, le Cyan entre le Bleu et le Vert dans le spectre des couleurs représenté par la décomposition de la lumière solaire à travers un prisme (prisme de Newton). Par le même principe et en combinant certaines des 3 couleurs fondamentales du spectre (Rouge-Vert-Bleu), ceci dans des proportions judicieusement choisies, on obtient du Blanc. Ce sont ces trois couleurs primaires qui sont utilisées en vidéo. Elles permettent par combinaison l'obtention de palettes de plusieurs millions de couleurs.

La Synthèse soustractive : c'est le principe du peintre qui peint son tableau sur une toile blanche et obtient sa palette de teintes par superposition et mélange de ses couleurs primaires. Si on superpose du Rouge, du Vert et du Bleu, on obtiendra simplement du Noir, contrairement à l'éclairage qui donne du blanc. On parle alors d'absorbtion des couleurs autres que le Rouge, le Vert et le Bleu superposés. Pour retrouver ces couleurs fondamentales, il suffit d'extraire les couleurs complémentaires :

B+V = Cyan (complément du Vert)
R+V = Jaune (complément du Bleu)
R+B = Pourpre/Magenta (complément du Vert)
En superposant, suivant un dosage bien précis, les couleurs Cyan + Jaune + Pourpre, on obtient alors toutes les nuances du spectre.



Allure et temps du signal vidéo-composite

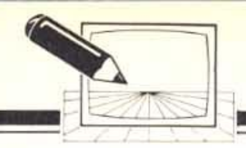
Voilà donc la grande différence entre Peinture et Lumière : on aboutit au même résultat avec une composition de base opposée. Chaque domaine aura encore ses propres couleurs fondamentales :
Imprimerie : Jaune-Magenta-Cyan.
Vision/vidéo : Rouge-Vert-Bleu.
Nous reprendrons plus en détail la colorimétrie lorsque nous aborderons la digitalisation couleur dans notre dossier multimédia, en utilisant de simples caméras Noir et Blanc pour constituer des images couleurs, nécessitant l'application de filtres colorés.

III) ***La synchronisation*** (synchro), qui tient lieu de chef et donne les ordres aux différentes parties du récepteur, de façon à reproduire le signal d'origine suivant une norme instaurée. Elle permet de fixer les limites gauche et droite de l'image sur l'écran (synchro horizontale -HD-), comme son début (haut) et sa fin (bas) (synchro verticale -VD-).

CRÉATION D'UNE IMAGE

C'est à partir des caractéristiques de l'oeil humain que s'établit un procédé complexe dans l'élaboration de chaque image. Il faut savoir qu'une image comporte 625 lignes horizontales successives. Le balayage s'effectue de la gauche vers la droite de votre écran par un "spot" (point de lumière) qui vient frapper le tube cathodique. L'empilement de ces lignes forme une image complète. Pour créer un effet de mouvement, il faut créer une succession d'images différentes les unes des autres avec une certaine vitesse, de manière à tromper l'oeil et donner l'effet de mouvement, à raison de x images par seconde. En télévision, ce nombre est fixé à 25 images par seconde. A ce stade interviennent donc 2 paramètres fondamentaux : 625 lignes et 25 images/seconde.

Pour que chaque ligne commence à gauche, finisse à droite et aille de haut en bas de l'écran, il faut encore de précieuses informations de synchronisation. Pour éviter le scintillement de l'image, dû à une vitesse d'écriture trop lente de chaque ligne sur l'écran, on a trouvé une astuce : elle consiste à entrelacer les lignes suivant le procédé pair/impair. On

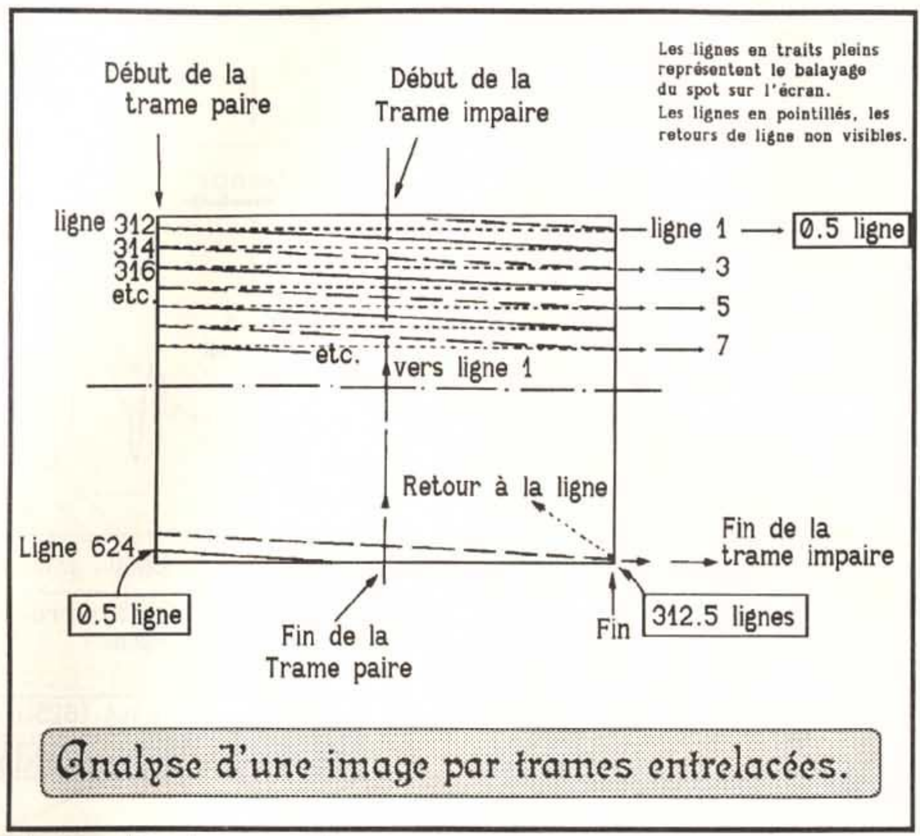


balaye d'abord l'écran une ligne sur deux, en commençant par les lignes impaires puis arrivé en bas de l'écran, on recommence le balayage en inscrivant les lignes paires. On divise alors 625/2 et on obtient 312.5 lignes. C'est ce qu'on appelle une trame ou demi-image. L'image comporte donc 2 trames différentes, dues au mouvement (25 images/ seconde ou 50 trames/ seconde). La trame est techniquement très utilisée dans les systèmes de digitalisation (Vidi-ST, par exemple) et les différents interfaces de mémorisation d'image (Mémoires de trame). Chili utilise aussi ce procédé, car c'est l'unique solution pour obtenir une image fixe et stable à partir d'un signal source vidéo en mouvement. Mais en parlant de trame, cela implique qu'on ne dispose que de 312.5 lignes, donc 2 fois moins d'informations et une perte de résolution, perceptible à l'oeil.

A la fin de chaque ligne, il faut que le spot puisse revenir et se placer au début de la ligne suivante. En aucun cas, on ne doit voir ce retour ; pour pallier ce problème, on affecte au spot, pendant le déplacement critique, un niveau inférieur au niveau du noir, donc invisible à l'oeil ; cela s'appelle le niveau de suppression présent à chaque ligne. Autre remarque non négligeable : parmi les 625 lignes disponibles, les 23 premières ne sont pas utilisées pour l'image, mais servent aux tests techniques : tops de codage (exemple Canal+) pour synchroniser l'électronique des récepteurs, et diverses fonctions (amplis/gain en fonction de la qualité de réception/ changement de niveau de codage pour les émissions à péage - télétexte - système Secam, etc.). Maintenant, un peu de calcul mental : l'écran ayant pour dimensions un rapport de 4/3 et disposant de 625 lignes, on obtient une résolution de :

```
625 x 4
------ = 830 points par ligne.
   3
```

Voilà, décrites à grands traits, les bases assez complexes mais indispensables de la vidéo. Pour vous aider, des diagrammes illustrent ces quelques explications ludiques et sim-



Analyse d'une image par lignes

plifiées. Que les passionnés se rassurent : il existe toute une panoplie d'ouvrages traitant dans le détail (à grand renfort de théories) chaque point des principes précédemment décrits.

NORMES ET DIVERSITÉS

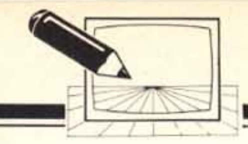
Tout cela serait encore trop simple, si les standards ne différaient suivant les pays, alors qu'aucun accord mondial n'a permis d'établir une norme commune sur la composition du signal vidéo : pourquoi faire simple quand il est si délicieux de se compliquer l'existence. Suite à certaines divergences techniques de conception, une panoplie de standards se sont fait jour, ne manquant pas de proliférer en tous sens sur l'ensemble du globe (PAL/ SECAM/ NTSC). Chose plus étrange, on aurait pu s'attendre à ce que le phénomène s'éteigne avec l'abandon des supports de communication analogiques, pour démarrer sur des bases saines la nouvelle ère numérique : il n'en est rien, la course entre le Japon et l'Europe pour le projet de télévision haute définition (TVHD) en fournit le plus parfait exemple.

Heureusement pour nous, par-delà ces malversations de standards et de formats, le principe théorique de la chose reste inchangé ; les différences se situent au niveau du codage de la couleur (PAL/ SECAM/ NTSC), et de la durée, différente suivant les standards, des tops de synchronisation. Nous y reviendrons en temps et en heure, dans des applications bien précises au fil de notre progression dans le domaine de la vidéo appliqué au micro-ordinateur.

Pour des raisons de technique et de qualité, le signal vidéo existe sous plusieurs formes :

- Signal composite N/B = Luminance
Quoi ? Mais oui, le noir et blanc existe encore !

- Signal composite Couleur = Luminance + Chrominance
Sur le même câble véhiculent toutes les informations. C'est le système le plus économique et le plus répandu, mais d'une qualité restreinte, destiné au "Grand Public". La qualité passable est due aux circuits de codage et décodage qui doivent s'occuper d'extraire toutes les informations qui



voyagent ensemble à l'arrivée (Synchros/ Luminance/ Chrominance), d'où une perte de qualité importante.

- Signal Y/C = Chrominance ET (exclusif) Luminance

...C'est la solution qui vient progressivement remplacer le signal vidéo composite, avec 2 informations véhiculées indépendamment et traitées individuellement. Ce procédé permet d'augmenter considérablement la qualité de l'image ; un circuit traite la luminance, un autre la chrominance.

- Signal RVB ou RGB =
Voie Rouge + Luminance
Voie Verte + Luminance
Voie Bleue + Luminance
Synchro HD/VD (ds certains cas).

...Le must en matière de vidéo. Utilisé par les professionnels de l'imagerie, c'est la solution adoptée par la plupart des fabricants d'ordinateurs (dont celui qui nous occupe), souvent pour des raisons de prix de revient. Regardez les composants manquants auprès du shifter de votre ST: encodeur et modulateur sont absents; car les développeurs étrangers du ST ont estimé que l'adaptation des ST pour le système Secam français n'était pas rentable... Ils nous ont quand même laissé le RVB, heureusement, car ce sont les signaux de base délivrés par le shifter. Réjouissons-nous : nous en tirerons avantage dans nos applications.

INTERFACES

D'après la diversité des signaux disponibles pour chaque type d'appareil, il devient nécessaire de pouvoir interconnecter tout matériel vidéo, de manière à constituer des ensembles cohérents et compatibles entre eux (genlocks, numériseurs, magnétoscopes, caméras S-VHS, etc.). Nombre de fabricants se sont penchés depuis peu sur ce type d'interfaces, lesquelles commencent à fleurir sous toutes formes et standards. Le premier exemple que nous allons traiter utilise les différents types de signaux traités précédemment : c'est la carte multimédia vidéo digitale CHILI.

Suite page 160

CHILI ET VIDÉO DIGITALE

S'il y avait un domaine où le ST restait encore dans l'ombre, c'était bien celui de la vidéo. Souffrant de cette carence depuis son début, devancé par ses concurrents, il fallait un miracle pour combler cette lacune : or voici qu'apparaît sur le marché une carte numérique vidéo plutôt fantastique, regroupant un nombre impressionnant de fonctions à faire pâlir un coloriste. Développée en Suisse par Marvin Ag. et distribuée par ALM en France, cette carte ouvre les frontières en matière de traitement numérique du signal vidéo et renforce l'image de marque du Mega ST.

COMPOSITION

Chili se compose d'une partie Hardware à connecter sur le bus d'extension d'un Mega ST2 ou 4, et d'une panoplie de logiciels (6 disquettes) contenant le système d'exploitation et les divers programmes :

a) Configuration de l'écran de visualisation ;
b) Dessin haute résolution en couleurs (65.536 couleurs affichables simultanément à l'écran) avec numérisation d'image et genlock ;
c) Création d'effets les plus divers : rouleau/rideau... ;
d) Création de séquences numériques ;
e) Banc de montage et Titrage ;
f) Lecture des séquences après montage ;
e) Programmes divers, tels que l'exportation des images vers les logiciels Retouche et Retouche PRO sous format .TIF ;
f) Sources divers livrés en langage C donnant la possibilité aux programmeurs de créer leurs propres applications et effets vidéo. A noter l'utilisation presque obligatoire du Turbo-C édité chez Borland International à cause de la bibliothèque utilisée par les programmeurs de Chili.

A QUOI ÇA SERT ?

Si vous êtes passionné de vidéo et fanatique de montages à partir d'un caméscope, ou encore un Pro qui souhaite intégrer son Mega ST dans sa configuration vidéo, vous êtes quand même invités à prendre connaissance de la suite : un endoctrinement éhonté sur le phénomène multimédia.
Chili est une carte vidéo intégrant des fonctions très diverses ne figurant jusqu'à présent que dans des systèmes vidéo très évolués tels que les régies professionnelles. Voici ce qu'offre Chili :

a) Une carte graphique Haute Résolution Couleur permettant un affichage simultané de 65536 couleurs, accompagnée d'un puissant logiciel de dessin intégrant un module de retouche d'images évolué. C'est actuellement une innovation dans le monde ST.

b) Un digitaliseur vidéo travaillant avec 256 niveaux de gris, 65536 couleurs en temps réel. C'est aussi le premier en son genre. Il permet la digitalisation en Trame ou en Image entière (double Trame) et demeure compatible avec certains scanners, à partir du programme Chili-Paint.

c) Un genlock offrant la puissance de la technique numérique face à celle de l'analogique classique utilisée jusqu'à présent. Le mode d'incrustation se fait en plein écran (overscan), toujours avec la palette évoquée plus haut.

d) Un générateur de caractères permettant le titrage de vos séquences vidéo avec des effets les plus divers, comme le scrolling hard de l'image dans tous les sens et formes. Les fontes utilisées sont celles du GEM, d'où une bibliothèque de polices de caractères très volumineuse (1000 polices existantes). Chaque police est "customisable" selon le goût de l'utilisateur.

e) Un correcteur d'images améliorant la qualité de vos films et séquences. Réglage des composantes RVB, contraste, luminance, phase, etc.

f) Un séquenceur d'images permettant, grâce à la digitalisation d'une séquence vidéo, la création de programmes autonomes tels que la représentation du fonctionnement d'un piston dans la culasse d'un



1 rue Pierre Dupont – 93200 Saint-Denis
Tel : 42 43 36 95 – Fax : 42 43 36 95

Une gamme unique de produits pour Atari®



ALM, un logiciel d'avance

BIONET 100

Réseau Ethernet alliant puissance et simplicité

Éthérogène (ST® , TT® , MS-DOS® , Unix® , Novel® , ...)

CALAMUS SL

L'aboutissement en matière de PAO

Modularité, souplesse et puissance sont les mots clés de ce logiciel.

Séparation couleur, justification verticale, import d'images professionnelles

RETOUCHE PRO

L'outil le plus évolué en matière de reprographie

Gère les images 256 niveaux de gris

Travaille en mémoire virtuelle (16Mo par image)

CHILI

Une carte genlock vidéo offrant une haute résolution, 65000 couleurs simultanées et un plein écran

De très nombreux logiciels livrés (titrage, dessin...)

UIS III

L'utilitaire indispensable

Un sélecteur d'objets autorisant toutes les manipulations (copie, déplacement...) à tout moment

DALI 4

"L'idée du siècle" (Atari Magazine)

Le logiciel de dessin le plus complet foctionnant sur ST, STE, Mega ST, TT

DIDOT LINEART

"Le plus abouti des logiciels de dessin vectoriel pour Atari" (SVM)

Vectorisation d'images scannées

SUPERCHARGER

L'émulateur PC le plus intéressant grâce à sa possibilité de bascule entre ST et PC

"Lequel acheter ? [...] une préférence pour le Supercharger..." (Guide d'achat ST-MAG)

SHERLOOK

La reconnaissance de caractères en pleine puissance.

Huit polices différentes, dictionnaire d'exception, automatisation du processus de reconnaissance...

SYNTEX

La reconnaissance de caractères en toute simplicité

Fonctionne comme accessoire

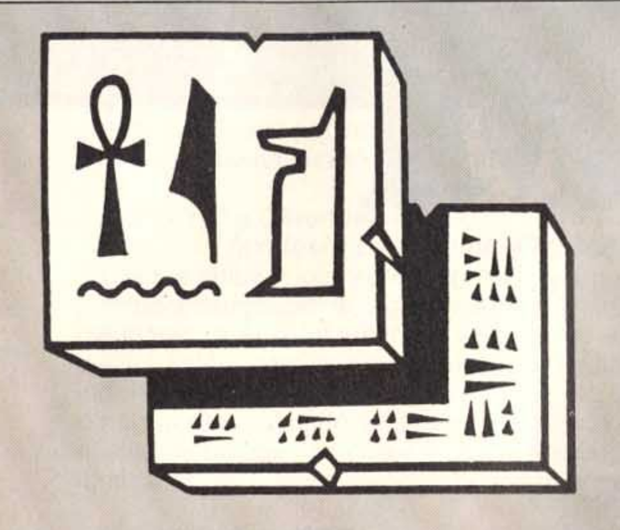
ICD

Cartes DMA-SCSI intelligentes

Cartes accélératrices. "C'est la carte la plus sophistiquée qu'il nous ait été donné de voir à ce jour" (ST-MAG)

ALM vous propose son catalogue complet de fontes Designer pour Calamus (plus de 70 typographies) pour **1490 frs**. Pour toute commande, contactez-nous.

Designer DESIGNER DESIGNER Designer Designer DESIGNER



DEVPACST2 EN FRANÇAIS

Depuis plusieurs mois déjà, nous vous avions incité, dans le cadre de l'initiation à l'assembleur, à utiliser le DevpacST2 de HiSoft pour la mise en oeuvre des programmes proposés. Ceux pour qui l'apprentissage de l'assembleur devait s'accompagner de la maîtrise de l'anglais sont désormais soulagés : la version française de ce remarquable outil de développement est bel et bien là. Par rapport à la version anglaise 2.22, testée dans le numéro 21 de ST MAG (Juillet 88), les améliorations apportées sont mineures mais pas encore superflues.

FAISONS LE POINT

Examinons tout d'abord le contenu de (l'unique) disquette du package de développement. Outre les deux principaux utilitaires : GENST2, éditeur/assembleur utilisant GEM, et MONST2, débogueur (en version TOS et en version GEM), on y trouve un débogueur résidant (AMONST) ; un éditeur de liens au format GST (LINKST) ; un compilateur de menu GEM (MENU2ASM) ; des exemples de programme en code source (utilisation du TOS, de l'AES, de la VDI et du compilateur de menus GEM) et quelques utilitaires supplémentaires (programme de diagnostic, inhibiteur de mode trace, correcteur d'environnement AES et sélecteur de fichier).

L'éditeur/assembleur est la pièce maîtresse de DevpacST2. C'est effectivement à partir de son menu que l'on peut lancer l'assemblage, le déboguage ou l'exécution d'un programme (en mémoire ou sur disque). Pour ne parler que des fonctionnalités d'édition, signalons qu'elles sont classiques : manipulation de blocs, recherche sélective, fusionnement de fichiers, etc. Les raccourcis clavier (pour une fois intelligents) permettent d'oublier complètement cette satanée souris et de gagner du temps. Ceci dit, en dépit d'une amélioration sensible, le scrolling n'est pas encore des plus rapides.

Le macro-assembleur fait partie de GENST2, toutefois une version '.TTP' est incluse dans le package afin de permettre le lancement de l'assemblage (sans édition) à partir d'un shell ou d'un environnement de programmation du type MENU+.



Outre la directive INCBIN d'inclusion de fichiers binaires déjà présentée lors du précédent test, notons encore l'amélioration de la vitesse d'assemblage grâce au cache disque intégré et la possibilité de produire des fichiers relogeables au format GST ou DRI (déjà depuis la version 2.0).

Toute l'originalité de DevpacST2 réside dans son débogueur qui, à sa première utilisation, déroute complètement l'utilisateur, mais dont la puissance n'est toujours pas égalée par ses concurrents ! Dans un environnement tel que celui de GEM où tous les courants d'air sont permis, autrement dit où il est possible d'ouvrir des fenêtres grandes comme des baies vitrées, on peut se demander pourquoi HiSoft s'est fatigué à se fabriquer "ses siennes" au look somme toute primaire... La réponse est simple : frimer ou déboguer, il faut choisir ! Qui ne connaît pas ces superbes débogueurs multifenêtres, pleins d'icônes où la souris est utilisée plein pot ? et par la même occasion GEM, l'AES, la VDI et toute la clique... Lorsque le débogueur utilise GEM en même temps que le programme à tester, bonjour les facilités de déboguage...

Ainsi MONST2 utilise ses propres "fenêtres", fabriquées avec ses propres routines de bas niveau, forcément c'est sommaire, sobre, accessible uniquement par des commandes clavier mais très efficace. Tout GEM est à disposition de votre programme, qui plus est, un basculement d'écran Débogueur/Application permet de voir, en pas à pas par exemple, l'effet de chaque appel GEM (AES ou VDI) sur le fonctionnement du programme d'application. Avec la documentation maintenant en français, vous n'avez plus aucune excuse pour ne pas utiliser DevpacST2 d'autant plus que dans un prochain épisode de l'initiation à l'assembleur (voir nos "Travaux Pratiques" en cahier central), nous vous proposerons un exemple d'utilisation complet, de l'édition au déboguage en passant par l'édition de liens pour ceux que cela intéresse (les divers fichiers '.BIN', '.LNK' à mettre en oeuvre).

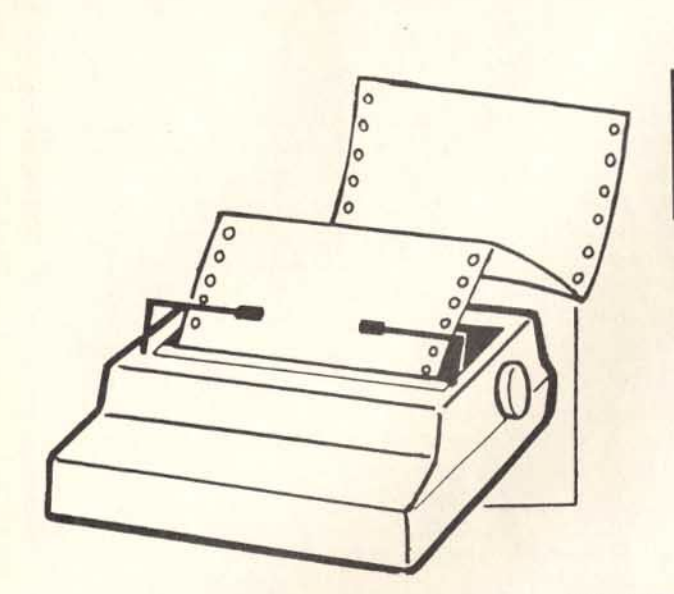
Pour en revenir à la documentation, pratique à maintenir ouverte devant le clavier (classeur 1/2 A4 de 180 pages), soulignons que la traduction nous change de certains autres logiciels où un mot est approximatively remplacé par un autre. En ce qui concerne le contenu, pas grand-chose à dire sauf peut-être sur la documentation des appels "système". En effet, si l'interface des fonctions GEMDOS, BIOS et XBIOS est détaillée, les appels aux fonctions GEM (AES et VDI) sont plutôt succintement examinés (et je vous entends me rétorquer que ce n'est pas le but d'un tel document... Soit, mais quand même !). Néanmoins, la présence sur disquette des fichiers "source" des macros d'appels à l'AES et à la VDI pallie en partie cette déficience (de plus, une bibliographie intelligemment sélective pour les débutants est donnée) et permet de recoller tous les morceaux pour ceux qui désirent appeler directement ces sur-couches logicielles du système d'exploitation par l'instruction "TRAP #2". Quoi qu'il en soit, l'indispensable (j'entends la documentation sur le débogueur et non sur les instructions du 68000 que l'on connaît par coeur depuis la sortie du Collector's sur le sujet) est là, c'est l'essentiel... J'allais oublier : un bonus pour la bafouille sur les "trucs et astuces de déboguage" !

Que souhaiter pour la version 3.0 ? Rien de plus pour le débogueur : il est bon ! De même pour l'assembleur : il est rapide et suffisamment conforme aux standards Motorola. Pour l'éditeur, pas grand-chose si ce n'est la possibilité d'ouvrir plusieurs fenêtres (éditions multiples) et des fonctionnalités de positionnement de marques dans le texte (le "goto line" nécessitant un effort inhumain de mémorisation du numéro de ligne). Oui, bien sûr, on pourrait y rajouter tout le superflu : sauvegarde automatique, "help" en ligne sur les instructions et, en cas de problème, composition automatique du numéro de téléphone de Human Technologies (distributeur de la chose dans l'hexagone). Mais bon...

Pour terminer, notons que depuis la première version de DevpacST (v0.50) jusqu'à cette dernière mouture (v2.25f), cinq années se sont écoulés ! Rares sont les produits dont la pérennité s'avère aussi forte; peut-être bien que celui-ci a su d'emblée séduire les amateurs et fidéliser les professionnels... Ajoutons enfin que la France Profonde (entendez par là, ceux qui refusent l'assimilation de la langue de Thatcher depuis la défaite de Waterloo) peut dorénavant goûter aux joies de la programmation assembleur sans effort linguistique grâce à cette version française tant attendue et réussie d'un produit déjà bien rodé.

C. PASCALADA





LE RÉDACTEUR VERSION 3.10

EPIGRAF ne nous a pas laissé trop attendre les nouveaux développements du Rédacteur. Cette nouvelle version ne se contente pas de corriger quelques petits bugs ou d'ajouter de petites options. Ce sont deux nouveaux modules complets qui viennent étendre ce monument : un gestionnaire de fichier (AZthèque), et une fonction de publipostage ("mailing").

LES CORRECTIONS

Le Rédacteur est sur le point de perdre sa mauvaise réputation de "bombe" des traitements de texte. En premier lieu, il effectue des sauvegardes automatiques de votre fichier avant un plantage (selon 3 critères) et lors d'un plantage. Néanmoins, dans ce dernier cas, l'opération ne s'avère pas fiable à 100%, puisqu'il nous est arrivé d'obtenir un fichier de 0 octets. Cependant, globalement, les plantages deviennent rares, et une nouvelle version encore plus fiable verra sûrement le jour.

L'ASSISTANCE

Un logiciel n'est pas uniquement constitué d'une disquette et d'un manuel. Il est également constitué d'un support technique qu'il soit épistolaire ou téléphonique. Et là, je crois que l'on peut tirer un coup de chapeau à Epigraf. Lorsqu'une nouvelle version est prête, tous les acquéreurs du Rédacteur ayant renvoyé leur carte d'enregistrement sont immédiatement informés individuellement par courrier. S'ils le désirent, ils reçoivent sous 10 jours la nouvelle version pour la modique somme de 30F (support magnétique, emballage et port compris).
Mais la palme revient à l'assistance téléphonique. Et je parle en connaissance de cause pour y avoir eu recours "incognito" par trois fois. Vous pouvez facilement entrer en dialogue avec un développeur affable vous donnant des informations qui vous auraient échappé en lisant le manuel. Vous pouvez également, et je vous y encourage vivement, signaler tous les bugs que vous avez pu rencontrer : vous êtes certains qu'ils disparaîtront lors de la prochaine version. Pendant que vous y êtes, vous pouvez demander quelques petites nouvelles options... ça marche ! Bien sûr, téléphoner à Toulouse n'est pas donné. Mais quelques francs, ce n'est pas cher payé si l'on songe que l'on va entendre l'accent chantant de ces sympathiques occitans ! J'invite donc tous les insatisfaits à noter précisément tous leurs griefs et à téléphoner au numéro indiqué sur le manuel (aux heures réservées !).

OCCUPATION DE LA MÉMOIRE

Que les possesseurs de ST ne possédant qu'un méga-octet de mémoire se rassurent : le Rédacteur 3.10 fonctionne plus que jamais sur leur ST et il en sera probablement ainsi pour les versions ultérieures. De plus en plus, le code correspondant à des options peu souvent utilisées (comme les différentes configurations) est en "overlay", c'est-à-dire qu'il est chargé depuis la disquette seulement lorsque l'on en a besoin. Cependant, le programme de paramétrage, appelable directement depuis le programme principal, risque de ne pas être exécutable au cas où l'on travaille un document important dans un contexte comprenant 1 Mo de mémoire, le dictionnaire et quelques utilitaires en dossier Auto, du type Turbo ST et TurboDos.

Que les possesseurs de Mega ST, TT et autres "GIGA-TT" se rassurent

aussi : il est possible de conserver toutes les options résidentes en mémoire vive. Personne n'a donc été oublié.

L'ADDENDUM DU MANUEL

Fini la grandiloquence, l'heure est à l'efficacité. L'addendum ne comporte que 36 pages bien tassées et va droit à l'essentiel, peut-être même un peu trop. Par conséquent, n'oubliez surtout pas de lire les deux fichiers AZTHEQUE.LIB et MAILING.LIB vous fournissant les errata et addenda à l'addendum. J'espère que l'Addendum sera réimprimé pour ne pas avoir une documentation répartie entre le manuel, son addendum et une disquette, faute de quoi tout cela risque d'être un peu trop déroutant pour les nouveaux acquéreurs.

AZTHEQUE

Ce module constitue vraiment un programme à part qui peut être appelé directement à partir du bureau GEM, et il est remarquable pour sa souplesse et sa facilité d'utilisation. Il s'agit principalement d'un gestionnaire de fichiers, une base de données non relationnelle. La notion de lien ou de parenté n'existe pas, toutes les fiches sont donc au même niveau. AZthèque ne prétend pas concurrencer Superbase, mais convient parfaitement pour archiver vos livres, vos cassettes vidéo ou vos étiquettes d'emballage de fromage.

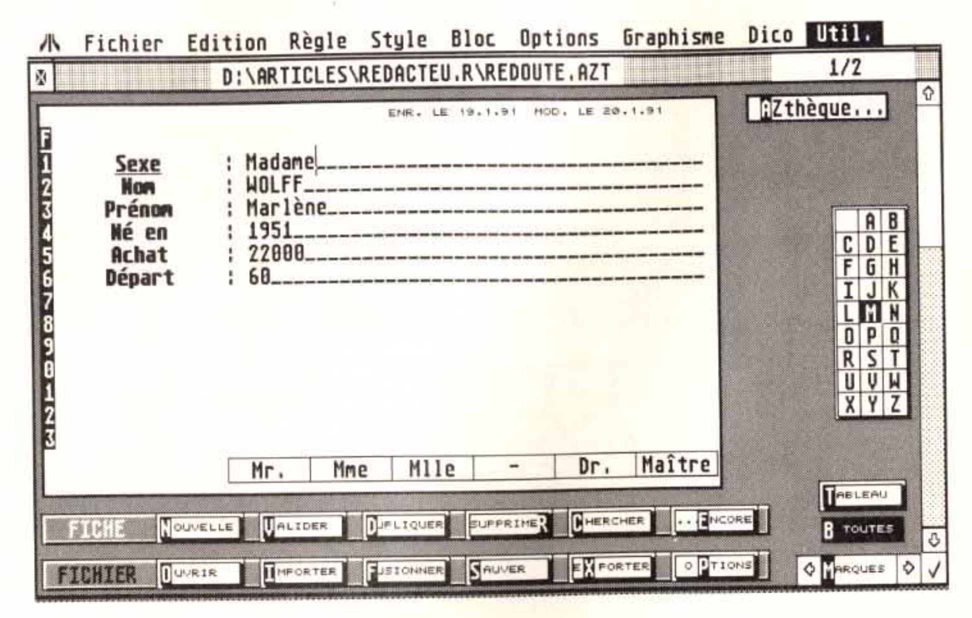
Vous pouvez facilement constituer et modifier le masque de saisie de vos fiches pouvant comprendre jusqu'à 13 champs que vous pouvez découper en 13 sous-rubriques. Les champs peuvent être de 4 types différents. Un type particulier mettant automatiquement en majuscule la première lettre du champ s'avère très pratique (prénom d'une personne, titre d'un film, etc). La saisie est facilitée par l'existence de quatre presse-papiers. Vous pouvez effectuer un marquage des fiches et des tris selon n'importe quel champ. L'importation de fichiers de type "Superbase", "Emulcom 3", "Works", "Portofolio" et "Rédacteur 3" est possible. L'exportation n'est en revanche possible que sous forme de fichiers ASCII ou vers le tampon "Undo" du Rédacteur 3, et il n'est pas possible d'imprimer directement (le contenu est cependant récupérable sous Rédacteur pour mise en forme et impression).

Sans être aussi génial que le Rédacteur 3, AZthèque constitue une bonne petite base de données mais son achat n'est pas justifié si vous possédez déjà une autre base de données. L'intérêt d'appeler AZthèque à partir du Rédacteur 3 n'est à mon avis qu'assez limité sauf en phase d'essais de publipostage. Comme vous ne pouvez acquérir le module de publipostage sans AZthèque, la question de l'achat repose surtout sur votre besoin de publipostage.

LE PUBLIPOSTAGE

Le mot "publipostage" vient des mots "publicité" et "postage" et signifie "prospection publicitaire ou vente par voie postale" ("mailing" en anglais). Nous recevons tous





des lettres qui semblent être écrites de la main même du PDG d'une grande société de distribution qui nous porte particulièrement dans son coeur. Voilà l'image négative du publipostage ; mais vous pouvez l'utiliser à des buts non lucratifs et même amicaux.

L'utilisation du publipostage dans le Rédacteur est d'une simplicité enfantine ; en voici le mode d'emploi. En premier lieu, il faut vous procurer une liste de gens à qui envoyer votre courrier sous forme d'un fichier d'AZthèque. En second lieu, vous écrivez votre lettre "type" sous le Rédacteur.
Le menu "Options" comporte un nouveau sous-menu "Champs Fusion" vous permettant d'effectuer les "personnalisations". Ce sous-menu s'adapte à la base de données utilisée.
Là où vous voulez faire apparaître un des champs du fichier AZthèque, par exemple le champ "couleur", vous cliquez sur l'option du même nom : les caractères "«couleur»" s'insèrent dans votre texte. Si vous voulez que votre lettre soit très "personnalisée", alors vous utiliserez l'option "SI... ALORS..." que vous programmerez un peu comme en Basic : les caractères "«condition»" s'insèrent dans votre texte. Vous pouvez également demander à ce que l'heure ou la date de votre ST soit indiquée (insertion de "«date»" et "«heure»") sous tous les formats couramment usités en France et dans les pays anglophones.

Je ne résiste pas à l'envie de vous montrer ces deux exemples. Mon chronomètre m'indique que j'ai mis 38 minutes pour lire le manuel, créer les 2 bases de données et les 2 lettres types, générer les 4 lettres personnalisées et incorporer le tout dans ce texte.

PREMIER EXEMPLE :

La base de données :

Sexe: Madame
Nom: WOLFF
Prénom: Marlène
Né en: 1951
Achat: 22000
Départ.: 60

Sexe: Monsieur
Nom: FULL
Prénom: Emile
Né en: 1946
Achat: 800
Départ.: 93

La lettre type :

«condition» «Sexe» «Nom»,

J'ai l'honneur de vous annoncer que vous êtes l'heureu«condition» gagnant«condition» de notre grand concours doté de 1000000000 (de LIT). En effet, «Prénom» - permettez-moi de vous appeler «Prénom» -, c'est votre année de naissance («Né en») qui a été tirée au sort. Vous pourrez donc participer au tirage final et être parmi les 1000 gagnants de ce fabuleux trésor. Vous pourrez ainsi acquérir «condition» de vos rêves !

Les lettres personnalisées :

Chère Madame WOLFF,

J'ai l'honneur de vous annoncer que vous êtes l'heureuse gagnante de notre grand concours doté de 1000000000 (de LIT). En effet, Marlène - permettez-moi de vous appeler Marlène -, c'est votre année de naissance (1951) qui a été tirée au sort. Vous pourrez donc participer au tirage final et être parmi les 1000 gagnants de ce fabuleux trésor. Vous pourrez ainsi acquérir le vison de vos rêves !

Cher Monsieur FULL,

J'ai l'honneur de vous annoncer que vous êtes l'heureux gagnant de notre grand concours doté de 1000000000 (de LIT). En effet, Emile - permettez-moi de vous appeler Emile -, c'est votre année de naissance (1946) qui a été tirée au sort. Vous pourrez donc participer au tirage final et être parmi les 1000 gagnants de ce fabuleux trésor. Vous pourrez ainsi acquérir la voiture de vos rêves !

Evidemment, si vous êtes le PDG de "L'Escroquerie Redoutable", vous pourrez demander au publipostage de faire une campagne de charme limitée aux clients un peu réticents (ceux qui ont un champ "Achat" faible) ou aux bonnes vaches à lait ("Achat" important), ou encore, limitée à un département particulier.

DEUXIEME EXEMPLE :

La base de données :

Sexe: Mademoiselle
Nom: Weaver
Prénom: Sigourney
Yeux: verts

Sexe: Monsieur
Nom: Dean
Prénom: James
Yeux: bleus

La lettre type :

Paris, le «date»
«condition» «Prénom»,

C'est avec plaisir que je t'invite à mon anniversaire. Il est «heure» à

ma montre que je regarde constamment tant il me tarde de te revoir. N'hésite pas à revêtir «condition» qui va si bien avec tes yeux «Yeux».

Les lettres personnalisées :

Paris, le 19 Janvier 1991

Chère Sigourney,

C'est avec plaisir que je t'invite à mon anniversaire. Il est 18h 26' 57" à ma montre que je regarde constamment tant il me tarde de te revoir. N'hésite pas à revêtir ta robe de soirée qui va si bien avec tes yeux verts.

Paris, le 19 Janvier 1991

Cher James,

C'est avec plaisir que je t'invite à mon anniversaire. Il est 18h 26' 59" à ma montre que je regarde constamment tant il me tarde de te revoir. N'hésite pas à revêtir ton costume trois pièces qui va si bien avec tes yeux bleus.

Vous pouvez bien sûr n'envoyer cette invitation qu'aux personnes du sexe de votre choix... Se limiter à une certaine couleur de yeux serait d'un mauvais goût...
Vous pouvez enfin ajouter des champs "«saisie»". Lorsque vous lancez le publipostage, vous pourrez entrer un texte "ultra" personnel. Par exemple, "P.S.: n'oublie pas de me ramener ma cassette".

Une fois les lettres "personnalisées" imprimées, il ne vous reste plus qu'à demander à AZthèque d'imprimer les adresses sur des étiquettes que vous collerez sur les enveloppes (le module "Impression des timbres" n'est pas prévu).

Si, à l'usage, le publipostage est très agréable, des utilisateurs exigeants pourront lui reprocher un léger manque de puissance. La programmation des champs "«condition»" ne prévoit qu'une structure du type "SI champ... ALORS...", alors qu'on aurait souhaité un degré de complexité plus élevé (conditions sur plusieurs champs, plusieurs "SI...ALORS..." pour une même «condition», etc). Mais ce type de condition peut être réalisé dans le module d'exportation du gestionnaire de fichier utilisé, et il est somme toute assez rare d'avoir souvent recours à une telle complexité. Cependant, appuyer sur "Undo" après la suppression d'un "«xxx»" dans la lettre type, ne restitue qu'un malheureux petit "«". Après la programmation d'une "«condition»", il n'est pas possible de la modifier ni même de la revoir (ou du moins je n'ai pas réussi à le faire) et cela est TRES agaçant. Lorsque vous choisissez le fichier qui contiendra les lettres "personnalisées", l'option publipostage ne vérifie pas si ce fichier existe déjà, comme avec l'option "Sauver" du menu "Fichier" : on travaille sans filet, ce qui n'est pas habituel avec le Rédacteur.

UN ENVIRONNEMENT A LA CARTE

Les nouveaux modules proposés pour le Rédacteur ne sont pas inclus dans le prix de base (990F TTC). Il vous



```
Fichier  Edition  Règle  Style  Bloc  Options  Graphisme  Dico  Util.
                D:\ARTICLES   Glossaire...        ^G
LIGNE:    2  L/P:  2/64  COL:   Touches...         ^T     20/01/91  17:23:54
                              Paramétrage         ^&
           «condition»                                    Date Courante    ^D
                              Enregistrer macro           Heure Courante   ^H
     J'ai    l'honneur    de
l'heureu«condition»  gagnant«co  Analyse de texte ^Y      Fichier Fusion   ^F
doté de 1000000000 (de LIT). En  Occurrences      ^;      Saisie clavier   ^S
vous  appeler «Prénom»*-, c'est  Archivage...     ^P      SI...Alors...    ^I
qui  a  été  tirée au sort. Vou                           Sexe             ^1
final  et être parmi les 1000 g  Voir l'index...  ^V      Nom              ^2
pourrez ainsi acquérir «conditi  Gérer l'index... ^J      Prénom           ^3
                                 Tri des Paragraphes      Né en            ^4
                                                          Achat            ^5
                                 Champs Fusion            Départ           ^6
                                 Publipostage             Champ 7          ^7
                                                          Champ 8          ^8
                                 Césure                   Champ 9          ^9
                                                          Champ 10         ^0
                                                          Champ 11         ^A
                                                          Champ 12         ^B
                                                          Champ 13         ^C
                                                          En-tête          ^E
INDEXER  MAJ-min  INVERSE  RELIEF  INDICE  1 STYLE  LIG=BLC  SUP.LI
F1 GRAS  F2 SOULIGNE  F3 ITALIQUE  F4 LEGER  F5 EXPOSANT  F6 RESTYLE  F7 MOT=BLK  F8 SUP.BLK
```

faudra donc débourser 100F supplémentaires pour le gestionnaire de fichiers et le publipostage (le prix du Rédacteur complet passe à 1090F TTC). Cependant, les anciens acquéreurs des versions 3.0x ne désirant pas acquérir ces nouveaux modules peuvent pour une P.A.F. de 30F se contenter des nouvelles versions du Rédacteur, déboguées et légèrement améliorées.

Au chapitre "environnement", il nous faut aussi signaler certaines améliorations qui contribuent à un plus grand confort d'utilisation : il est possible, désormais, de charger une bibliothèque de règles, une liste de mots utilisateur ou une table de transcodage par défaut. Enfin, il est possible de modifier l'éditeur de dictionnaire pour travailler à partir d'une liste de mots.

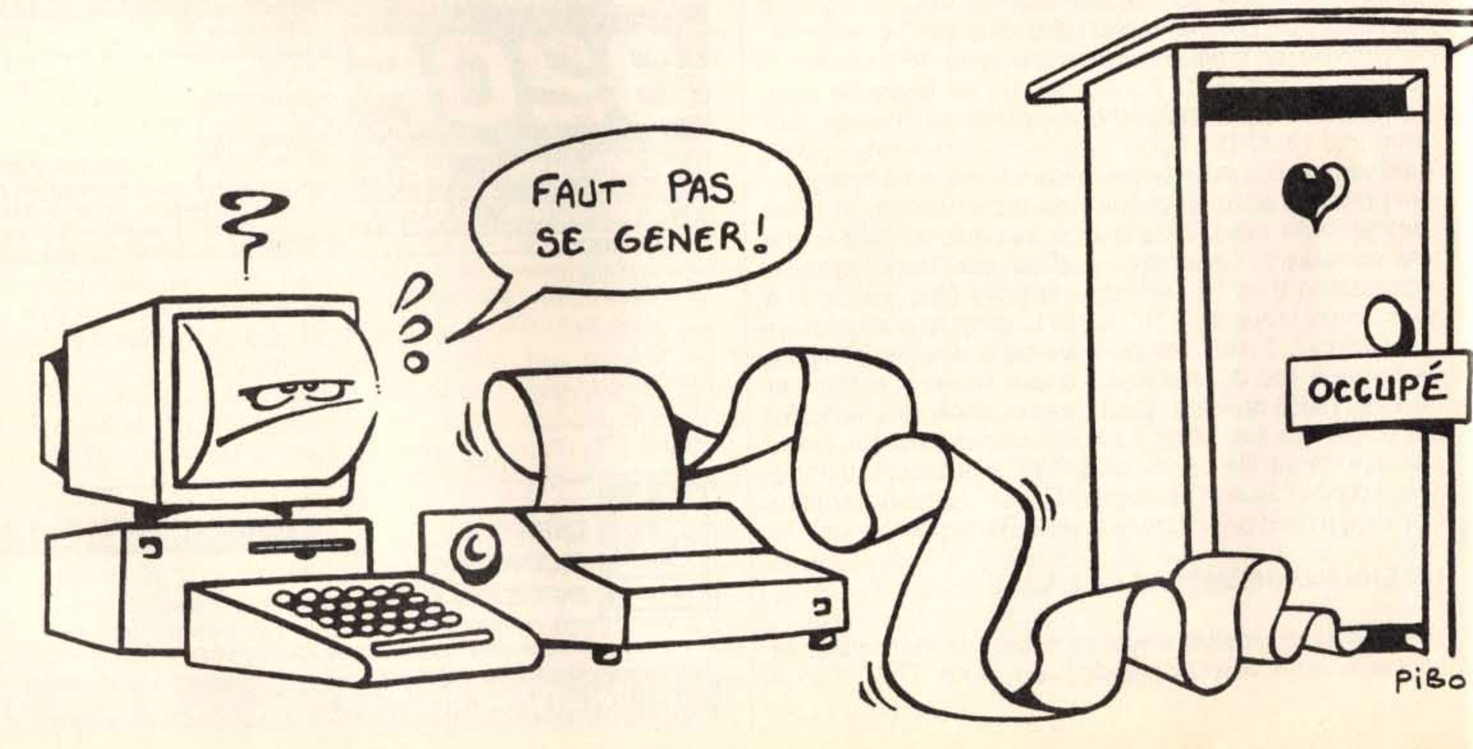
COMPLET ?

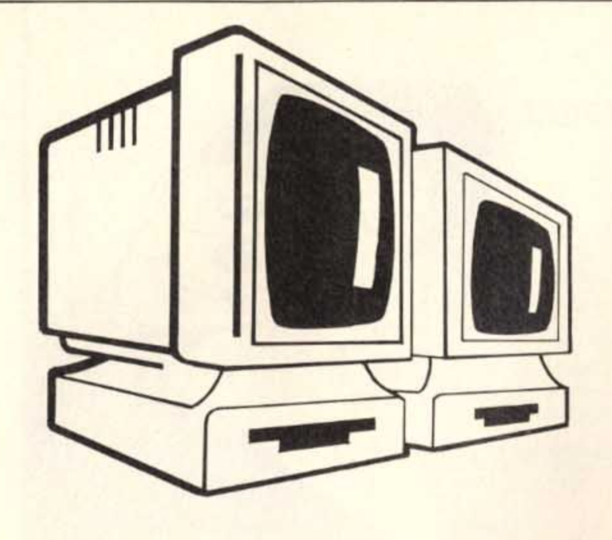
Mais que manque-t-il encore au Rédacteur ? Précisons d'abord que sa vocation n'est pas de devenir un intégré comme Framework (un logiciel pour PC) : traitement de texte, superbase de données, tableur et graphiques. Il propose en revanche une base de données non relationnelle, de nombreux utilitaires et des tableaux avec totaux calculés. Il n'offre pas de multi-colonnage ni de création de graphiques mais permet des hardcopy d'écran, importe des graphiques de tous formats et possède un éditeur de formules mathématiques puissant. Les sept disquettes double-face (soit vingt et un programmes) constituent un véritable environnement de rédaction, qui va connaître encore des extensions. Jugez-en plutôt à la vue des projets de développement échelonnés jusqu'en Mars : version développeur, dictionnaire anglais et gestion de la langue anglaise, dictionnaire des synonymes, polices complémentaires, gestion de plan. Il ne manquera plus que l'encyclopédie sur CD-ROM...

CONCLUSION

Le Rédacteur s'affirme encore plus comme une véritable tête d'affiche au sein des traitements de texte sur ST. Malgré quelques petits bugs persistants, je félicite les développeurs d'Epigraf car ils ont manifestement tenu compte de nos remarques. Comme quoi, si vous voulez devenir de grands programmeurs, lisez ST Mag !...

Luc Miotti





ST-RAM ET TT-RAM

Nous vous avions déjà glissé un mot sur les deux types de RAM qui équipent le TT, baptisées ST-RAM et TT-RAM. Nous rappellerons ici leurs caractéristiques, avant de parler de leur usage pour le programmeur, et l'utilisateur.

La ST-RAM a un usage rigoureusement identique à l'ensemble de la RAM du ST, c'est-à-dire qu'elle est partagée par l'ensemble des processeurs, en particulier le Shifter (celui-ci prend le bus exactement la moitié du temps, par tranches de 250 ns), et les traitements en DMA, ne laissant que relativement peu de temps au 68030 pour y accéder.

La TT-RAM, quant à elle, est entièrement dévolue au 68030 et aux deux nouveaux canaux DMA (SCSI et SCC), rendant son accès bigrement plus rapide, mais avec certaines contraintes d'utilisation, puisqu'on ne peut pas utiliser cette RAM pour des écrans alternatifs, ou pour faire du transfert DMA ACSI (disque dur à l'ancienne norme ou lecteur de disquettes), ou encore pour faire du son DMA.
Pour les utilisateurs habitués à d'autres systèmes, en particulier l'Amiga, la ST-RAM est l'analogue de la Chip RAM, et la TT-RAM pourrait être appelée Fast RAM.

PROGRAMMEURS, LISEZ CECI !

Voyons maintenant comment ceci se présente aux yeux du programmeur. Celui-ci doit d'abord déterminer ses besoins en RAM, c'est-à-dire s'il a besoin de ST-RAM ou pas (on ne peut pas avoir BESOIN de TT-RAM... On ne peut rien y faire qui ne soit pas possible en ST-RAM).
S'il n'a aucun besoin du tout en matière de ST-RAM, autant faire en sorte que la TT-RAM soit utilisée au maximum. Pour cela, il faut préciser que le programme ET la mémoire qu'il alloue peuvent être placés en TT-RAM. Il suffit de modifier deux bits de l'en-tête du programme à cet effet, les bits 1 et 2 (le bit 0 étant le bit de poids faible) du treizième mot, et de les mettre à 1.

Il est aussi possible que le programme ait besoin de ST-RAM, mais uniquement pour des blocs qu'il alloue, et non pour le chargement du programme lui-même. Dans ce cas, seul le bit 1 doit être positionné. Le cas inverse peut se présenter aussi, vous devinez ce qu'il faut faire dans ce cas.
A priori, seul le développeur sait de quel type de RAM il a spécialement besoin, et peut modifier son programme en conséquence. Mais en examinant attentivement le comportement d'un programme, il est possible d'effectuer la manipulation soi-même (avec la plus grande prudence, et en vérifiant in extenso que le programme supporte bien la modification - je me décharge de toute responsabilité en cas de problème dû à l'utilisation de cette méthode).

Encore un détail : le fait que les bits soient positionnés signifie qu'il est préférable d'utiliser de la TT-RAM, mais pas obligatoire. En cas de manque de place en TT-RAM, le TOS peut très bien utiliser de la ST-RAM.
Un cas encore plus délicat est celui où il y a encore de la TT-RAM disponible, mais plus de ST-RAM. La plupart des programmes utilisant une gestion dynamique de la RAM avec Malloc, pourront être chargés en TT-RAM, Malloc leur fournissant autant que possible de la TT-RAM, et quand il n'y en a plus, de la ST-RAM.

Mais d'autres programmes, en particulier ceux écrits en C Alcyon (pouah !) ou en GfA (repouah !), prennent le plus de place possible dans le programme lui-même. Dans ce cas, le chargement en TT-RAM, avantageux du point de vue de la vitesse, est par contre contraignant du point de vue de la place disponible. Atari a donc prévu pour cela, dans les 4 bits de poids fort du même treizième mot, un champ baptisé "TPAsize".

Il n'est donc utilisé que dans le cas où le programme peut être chargé en TT-RAM, ET qu'il y a moins de TT-RAM que de ST-RAM. Si ce cas se présente, le Gemdos prend TPA-size (qui peut varier de 0 à 15), lui ajoute 1, multiplie le tout par 128 ko, et y ajoute la taille déclarée du programme (segments TEXT, DATA et BSS). Si la valeur obtenue est inférieure à la quantité de TT-RAM disponible, on l'y charge, sinon on met le programme en ST-RAM. Ouf !

DISQUE DUR GOLDEN PREMIUM
A PARTIR DE 3750 Francs TTC

Prix en forte baisse : nos quantités d'achats augmentant fortement, nos fournisseurs ont baissé leur prix. Nous répercutons intégralement cette baisse afin de vous proposer les meilleurs prix.

Code Produit	Débit (en Ko/s*)	Temps d'accès (en ms)	Capacité formatée (en Mo)	Prix TTC
GP 40	680	19	40	3750,00
GP 52	950	17	52	3990,00
GP 80	680	19	80	5490,00
GP 105	950	17	105	6290,00
GP 210	680	19	210	8990,00

* Le débit est donné par le programme RATE HD. Il correspond à une utilisation réelle.

DISQUE DUR GOLDEN PREMIUM

- Mémoire cache de 64 Ko intégrée,
- Logiciel de formatage et de partitionnement fournis,
- Disque silencieux,
- Cable de raccord (DMA et alimentation)
- Interface SCSI/DMA haute vitesse,
- 100% compatible disques durs Atari,
- 100% compatible PC SPEED, Supercharger, Spectre...
- Boitier de taille réduite,
- Livré monté et testé (formaté, partitionné et auto-bootable),
- Sortie SCSI,
- Disque auto-parquable.

- **2 ans de garantie,**
- **Disque dur QUANTUM,**
- **Horloge intégrée sauvegardée par batterie,**
- **Livraison sous 24 h (dans la limite des stocks),**
- **Logiciel de défragmentation et de réparation fourni.**

Code Produit	Débit (en Ko/s*)	Temps d'accès (en ms)	Capacité formatée (en Mo)	Prix TTC
P 32+	600	28	32	3290,00
P 48+	600	28	48	3490,00
P 84	650	24	84	4490,00

* Le débit est donné par le programme RATE HD. Il correspond à une utilisation réelle.

DISQUE DUR PREMIUM

- Garantie 1 an,
- Logiciels de formatage et de partitionnement fournis,
- Cables de raccord (DMA et alimentation)
- 100% compatible disques durs Atari,
- 100% compatible PC SPEED, Supercharger, Spectre...
- Livré monté et testé (formaté, partionné et auto-bootable),
- Disque auto-parquable.

Nouveau !

PAIEMENT EN 4 FOIS*

SATISFAIT OU REMBOURSE**

* Le paiement en 4 fois sans frais est soumis à l'accord d'un organisme bancaire. Consultez-nous pour constituer votre dossier.

** Vous disposez de 2 semaines pour tester les disques durs GOLDEN PREMIUM. Consultez-nous pour plus d'informations.

OMIKRON S'AGRANDIT ET CHANGE D'ADRESSE !

OMIKRON.

7, Rue Voltaire / 51100 REIMS
Tél. : 26.40.60.22 / Fax. : 26.97.71.39
RC : 343.750.972.00014

Commande par Téléphone Possible !

BON DE COMMANDE

A remplir très lisiblement en capitales et à retourner à :

OMIKRON - FRANCE
7, rue Voltaire / 51100 REIMS
Tél. 26.40.60.22 / Fax. 26.97.71.39

NOM :
Prénom :
Adresse :
Code Postal : Ville
Règlement par chèque joint à la commande, par carte bleue, contre remboursement ou en 4 fois.
Par C.B.
Date de validité :

Pour paiement en 4 fois sans frais, joindre : 1 photocopie recto-verso de votre carte d'identité, une photocopie de votre dernier bulletin de salaire, un relevé d'identité bancaire ou postal et une attestation de domicile (Facture EDF, France Télécom, loyer...). Le dossier est ensuite soumis à l'accord de l'organisme bancaire.

Code Produit	Qantité	PRIX
Port disque dur : 100 F		
Contre remboursement : +60 F		
TOTAL :		

S 02-91



Je vous récapépète l'usage du treizième mot ? On y va :

```
15 14 13 12 11 10  9  8  7  6  5  4  3  2  1  0
<-TPAsize-> <-------réservé (à 0)------->  M  P  F

(TPAsize+1)*128 Ko + TEXT+DATA+BSS :
     quantité de TT-RAM en dessous de laquelle il vaut mieux prendre de la
     ST-RAM, s'il n'y en a plus.
F :  Fastload. Si à 1, on n'efface que BSS au chargement du programme, sinon
     toute la RAM disponible.
P :  Si à 1, préférence pour la TT-RAM pour le chargement du programme, sinon
     chargement obligatoire en ST-RAM.
M :  Si à 1, préférence pour la TT-RAM pour Malloc, sinon ST-RAM uniquement.
```

LE RETOUR DU FILS DE MALLOC

Tout ceci ne concerne que les programmes ayant été écrits avant l'apparition des deux types de RAM. Pour ceux qui seront développés à partir de maintenant, il existe des moyens de contrôler plus efficacement la RAM utilisée, en passant par la fonction Mxalloc du Gemdos, en lieu et place de Malloc. Tout d'abord, son appel :

En C ANSI, le prototype serait :

```
long Mxalloc(long combien,int mode);
```

Avec l'ajout de la fonction :

```
long Mxalloc(long combien,int mode)
{
   return gemdos(0x44,combien,mode);
}
```

Avec un autre C, une macro suffit :

```
#define Mxalloc(a,b)
gemdos(0x44,combien,mode)
```

Mais assurez-vous que "combien" est un long.

En Assembleur :

```
MOVE.W   #mode,-(SP)
MOVE.L   #combien,-(SP)
MOVE.W   #$44,-(SP)
TRAP     #1
ADDQ.L   #8,SP
' le résultat est alors dans D0
```

Le mode peut prendre les valeurs suivantes :

0 ST-RAM uniquement
1 TT-RAM uniquement
2 ST-RAM de préférence
3 TT-RAM de préférence

Suivant la position du bit 2 de l'en-tête du programme, un Malloc "normal" correspond soit à un mode 0 (bit à 0, pas de TT-RAM), soit à un mode 3 (bit à 1, TT-RAM de préférence).

Comme avec Malloc, si on passe -1 comme quantité de RAM demandée, on obtient la taille du plus grand bloc disponible de la RAM souhaitée, les modes 2 et 3 donnant la taille du plus grand bloc, les deux types de RAM étant confondus.

"EXTENSIONNABILITÉ"

Une gestion correcte des deux types de RAM au mieux de leur disponibilité est importante. Il faut que je vous précise les modalités d'extension des deux types de RAM. Tout d'abord, rien n'est possible immédiatement, les modules d'extension de RAM du TT étant tous équipés d'un "custom-chip", c'est-à-dire d'un composant propre à Atari, qui en a donc le monopole, et qui ne fournit pas encore de telles extensions.
La ST-RAM est constituée de deux bancs, le premier sur la carte mère, le deuxième sur une carte d'extension. Les bancs sont constitués de 16 puces de 1 Mbit sur 4 bits de large (cette structure est nécessaire pour la mise en oeuvre du bus 64 bits utilisé pour l'accès à la RAM par le Shifter), soit un total de 2 Mo chacun. Dès que les puces 4 Mbits (en 4 bits de large, précisons-le encore) seront à un prix intéressant sur le marché, il sera possible de remplacer la carte d'extension par son équivalent en 8 Mo, pour atteindre ainsi 10 Mo de ST-RAM.
La TT-RAM est constituée d'un seul banc, monté sur une carte d'extension. Ici, on n'a besoin "que" de 32 bits de large, et les modules (de 4 Mo) fournis par Atari sont en fait constitués de 4 barrettes SIP soudées sur une carte, accompagnées par le contrôleur mémoire (le fameux custom-chip) correspondant. Ici encore, il sera possible de passer à 16 Mo en utilisant des RAMs de 4 Mbits.

Pour vous donner une meilleure idée des combinaisons possibles, je vous ai dressé ci-dessous un petit tableau, avec une distinction entre les configurations possibles "de suite", et celles pour lesquelles il faudra attendre un peu :

ST-RAM	Banc 1	Banc 2	TT-RAM	Total	4 Mbits	Remarque
2 Mo	2 Mo	-	-	2 Mo	-	TT 030/2 standard
4 Mo	2 Mo	2 Mo	-	4 Mo	-	
10 Mo	2 Mo	8 Mo	-	10 Mo	X	
2 Mo	2 Mo	-	4 Mo	6 Mo	-	
4 Mo	2 Mo	2 Mo	4 Mo	8 Mo	-	TT 030/8 standard
10 Mo	2 Mo	8 Mo	4 Mo	14 Mo	X	
2 Mo	2 Mo	-	16 Mo	18 Mo	X	
4 Mo	2 Mo	2 Mo	16 Mo	20 Mo	X	
10 Mo	2 Mo	8 Mo	16 Mo	26 Mo	X	

Voilà. Le problème épineux des deux types de RAM du TT aura enfin été traité dans son intégralité. Pour finir, je vous dirais simplement qu'il existe un troisième moyen d'étendre un TT : la RAM VME, qui permet de le porter à 2042 Mo... avec un TT/X !

Ne manquez pas la suite de nos aventures avec le TT et ses performances, dans ce même numéro, peut-être même à la suite de ce présent article, sait-on jamais...

TT : LES PERFORMANCES

Passons maintenant à un autre sujet de réjouissance : les performances du TT. Lors de nos différents articles sur le TT, nous vous avons fait part d'une bonne partie des caractéristiques géniales ou décevantes du TT. Mais il s'agit de savoir quel est en fait le résultat final pour l'utilisateur, et donc de mesurer les performances réelles de la machine. Ce mois-ci, nous vous proposons donc les résultats d'une batterie de tests, assez instructifs...

QUICK INDEX

Le programme de test de performances le plus courant sur ST, c'est Quick Index, de Darek Mihochka (un Canadien...). On peut classer les tests en trois catégories :
- processeur ;
- entrées-sorties disque ;
-entrées-sorties texte et graphiques.

Nous ne traiterons pas ici les entrées-sortie disque, les disques durs des ST étant très variés, il est difficile de choisir une référence, et les accès disquette n'étant pas d'une importance capitale pour un ordinateur qui dispose d'un disque dur en standard.

Les tests ont été effectués dans plusieurs cas de figure, en mode VGA (640 x 480, 16 couleurs) ou monochrome (640 x 400), en TT-RAM ou ST-RAM (en modifiant les bits correspondants de l'en-tête du programme - voir l'article sur les deux types de RAM dans ce même numéro), et avec ou sans cache, permettant de constater des différences assez importantes. Les résultats sont des indices à comparer avec la valeur 100 pour un ST "normal", équipé du TOS 1.4, avec un moniteur monochrome.

En ce qui concerne les tests processeur, dont vous trouverez les résultats ci-dessous, le choix VGA/duochrome n'a pas d'importance, et nous trouvons donc seulement quatre cas de figure, et encore :

Cache:	Avec		Sans	
RAM:	TT	ST	TT	ST
Mémoire	714	497	271	158
Registre	827		312	197
Division	1024		859	768
Décalage	3534		2929	2050

Avec le cache, l'utilisation de ST-RAM ou de TT-RAM ne change rien, puisque le test est en fait une boucle sur un petit bout de code, qui, une fois chargé dans le cache, ne nécessite plus d'accès à la RAM (sauf dans le cas du test de mémoire, évidemment !).

Ces tests donnent donc le TT entre 1,6 et 35 fois plus rapide qu'un ST, suivant les applications (les décalages sont très nettement accélérés sur un 68030, qui dispose d'un système de "décalage à barillet", particulièrement performant). Avec le cache, la plupart des opérations sont accélérées entre 5 et 10 fois, ce qui est tout de même particulièrement sympathique.

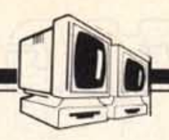
Voici maintenant les tests d'affichage, en mode texte (BIOS) et graphique (GEM). On rigole déjà moins. Comparons d'abord ce qui est le plus comparable, le fonctionnement en mode monochrome :

Cache	Avec	Sans
BIOS		
Text (ST-RAM)	229	167
Text (TT-RAM)	245	176
String	219	168
Scroll	297	207
GEM		
Draw	220	167

On constate donc un gain de l'ordre de 70 % à 110 % sans cache, et 120 % à 200 % avec (cas le plus courant). Rien de bien impressionnant, c'est même plutôt décevant. Mais ce n'est rien à côté de ce qui suit, c'est-à-dire les tests en mode VGA :

Cache	Avec	Sans
BIOS		
Text (ST-RAM)	139	77
Text (TT-RAM)	145	79
String	157	96
Scroll	67	48
GEM		
Draw	175	108

Comme vous le voyez, ici, on gagne très peu... quand on ne perd pas ! Le plus étonnant reste le rapport entre les vitesses en mode VGA et mode ST, qui reste le plus souvent de l'ordre de 2, montrant que ce n'est pas tellement dans l'affichage lui-même qu'il perd du temps (on



aurait une différence d'ordre 4, due au passage d'un seul plan de bits à 4), mais surtout dans les manipulations de diverses broutilles en RAM et en ROM.

À titre de comparaison, voici les résultats obtenus avec Turbo ST 1.82 (un accélérateur graphique logiciel, développé par SofTrek, et distribué en France par Arobace), sur un ST en TOS 1.4 avec Blitter :

BIOS	
Text	229
String	1287
Scroll	136
GEM	
Draw	286

Voici donc la phrase à méditer : "Un petit effort sur le logiciel vaut bien un gros effort sur le matériel." En général, ce n'est que lorsqu'on est arrivé à la limite du possible par logiciel que l'on améliore le matériel. Pas chez Atari. Dommage que Turbo ST ne marche pas sur TT...

DHRYSTONE & SIEVE

Mais retournons dans les tests de performance pure, basés essentiellement sur le calcul et la manipulation mémoire. Deux classiques au programme, Dhrystone (série d'opérations qui ne servent strictement à rien, mais savamment dosées pour correspondre à l'utilisation moyenne) et Sieve (crible d'Eratosthène, consistant à chercher les N premiers nombres premiers (ici N=8191)), compilés en Turbo C 2. Le tableau suivant donne les résultats en itérations par seconde et en indices par rapport au ST qui a 100 %.

L'idéal est évidemment de comparer la première et la dernière ligne, ce qui nous donne donc des rapports de l'ordre de 5 pour Dhrystone, et de 9 pour Sieve. Vous constaterez que Dhrystone profite autant de l'utilisation de la TT-RAM que du cache, alors que Sieve gagne surtout énormément avec le cache. Ceci est dû au fait que Sieve est basé sur une très courte boucle, au contraire de Dhrystone qui n'est qu'une grande boucle composée de nombreuses boucles plus ou moins petites, et qui ne tient pas toute entière dans le cache !

			Drhystone		Sieve	
ST			1574	100	5.0	100
TT	ST-RAM					
		sans cache	3168	201	9.9	199
		avec cache	4657	296	42.2	849
	TT-RAM					
		sans cache	5296	336	15.1	303
		avec cache	7836	498	45.7	919

ZZ-3D

Voici maintenant un test fourni par Human Technologies, basé sur une série d'opérations topologiques sur des objets tridimensionnels (soustraction de 12 cylindres du volume d'un cube) avec ZZ-3D. La version TT a été compilée en mode 030+882 avec le Lattice C 5, et vous avez droit, en prime, aux résultats de la même opération par le même programme sur un MicroVAX II (avec aucun autre programme, à part les tâches système, le tout sous VMS 4.6 avec 9 Mo de RAM) :

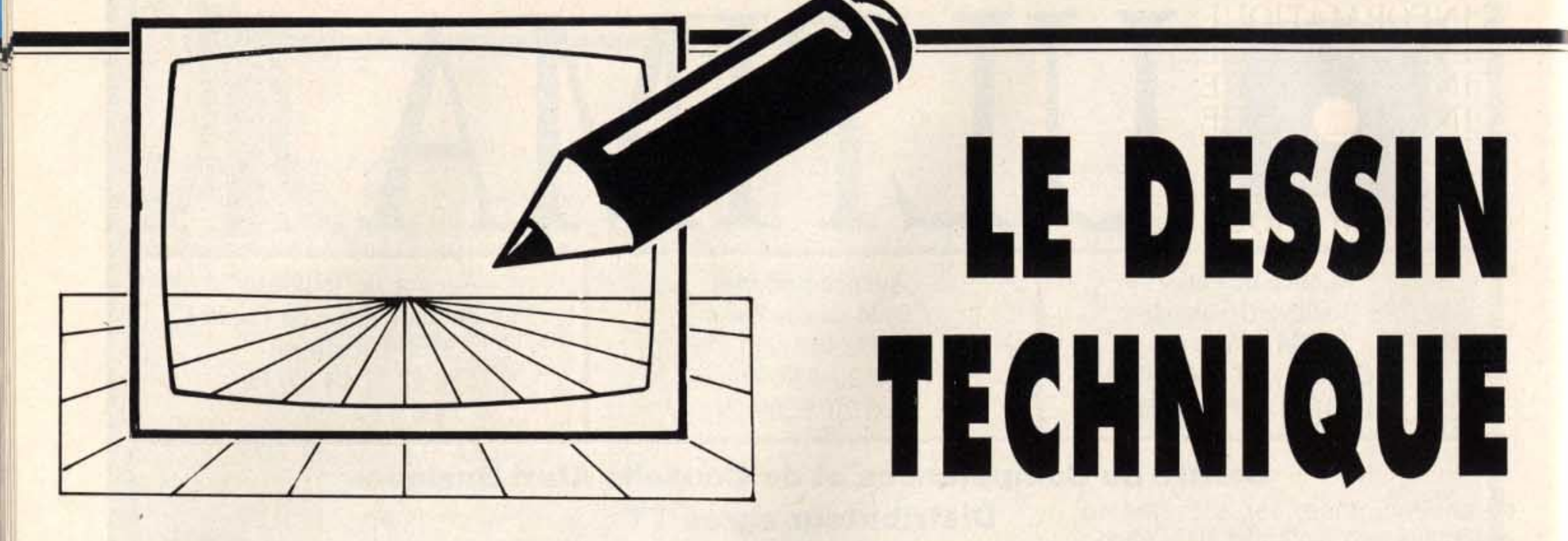
Opération	ST	ST+68881	TT 16 MHz	TT 32 MHz	MicroVAX II
1	6.84	3.22	0.75	0.53	0.96
2	16.99	6.28	1.56	1.04	1.80
3	27.37	9.38	2.38	1.59	2.82
4	37.89	12.70	4.25	2.13	3.84
5	116.52	40.73	9.59	6.70	12.47
6	137.42	51.78	14.43	8.21	7.38
7	154.98	63.12	15.05	10.00	21.28
8	167.04	74.02	17.60	12.43	28.22
9	172.71	92.16	20.74	14.10	34.43
10	188.91	102.16	23.48	16.70	39.78
11	201.66	113.90	26.21	18.46	44.58
12	210.34	127.20	29.03	20.23	50.91
Moyenne	119.89	58.05	13.59	9.34	21.54

Ce test, qui a l'avantage d'avoir aussi été effectué sur un TT à 16 MHz (la première version de la machine, présentée en 89 à Düsseldorf, et qui n'a été vendue qu'à des développeurs), nous rassure sur les performances de la version 32 MHz. Même si l'on est loin de doubler la vitesse de la machine, les performances sont quand même améliorées de 50 % environ.

Il me semble malgré tout qu'on se trouve ici dans un cas de figure favorable, puisque le test fait appel principalement à des manipulations de nombres flottants, qui sont ici traitées par le 68882, dont la vitesse a été doublée. Dans le cas de programmes plus orientés sur la manipulation de RAM, l'amélioration des performances doit être largement moins importante, surtout en ST-RAM.

Voilà la fin de ce petit tour des performances "brutes" du TT. Reste encore à voir ses performances globales, à l'usage. Mais il faudra encore quelque temps pour les mesurer pleinement. Je citerai quand même le Mic Dax de service : "Qu'est-ce que ça va être quand on va me le reprendre..."

Jacques Caron



LE DESSIN TECHNIQUE

Les logiciels de dessin technique 2D ne sont pas légion sur ST. Nous vous présentons ce mois-ci "Le Dessin Technique", édité par la société toulousaine Logisoft, qui risque de devenir un challenger de poids, en raison de son langage intégré, BASICAD, un Basic dédié au dessin.

LE DESSIN TECHNIQUE 2D

Les dessinateurs techniques font de plus en plus souvent appel aux méthodes informatiques pour simplifier leur travail. Mis à part ZZ-2D sur ST, édité par Human Technologies, on trouve généralement deux catégories de logiciels 2D : les logiciels d'accès facile, mais généralement peu puissants, et les poids lourds, type AUTOCAD sur PC et maintenant Macintosh, pesants à manipuler, mais comportant un langage interne de programmation destiné à simplifier encore la tâche du dessinateur en automatisant certaines opérations. Le Dessin Technique tente d'en faire un compromis.

L'OUTIL DE DESSIN

La boîte à outils de ce logiciel, assez complète, assure d'abord la génération des entités les plus courantes : segments, rectangles, triangles, cercles, ellipses et arcs. Or, le nombre d'options et de paramètres de création est également important : il est par exemple possible de créer un cercle soit avec son centre et son rayon, soit avec deux points sur un diamètre, soit encore avec trois points sur sa circonférence. Les opérations s'effectuent simplement en sélectionnant l'outil désiré dans la "toolbox", présentée sous la forme d'une fenêtre remplie d'icônes que l'on peut déplacer sur l'écran, de manière à diminuer les déplacements de la souris. La première zone de cette fenêtre correspond au menu principal et concerne le choix de l'opération à effectuer, création d'un objet, modification, copie de blocs, cotation, hachurage...

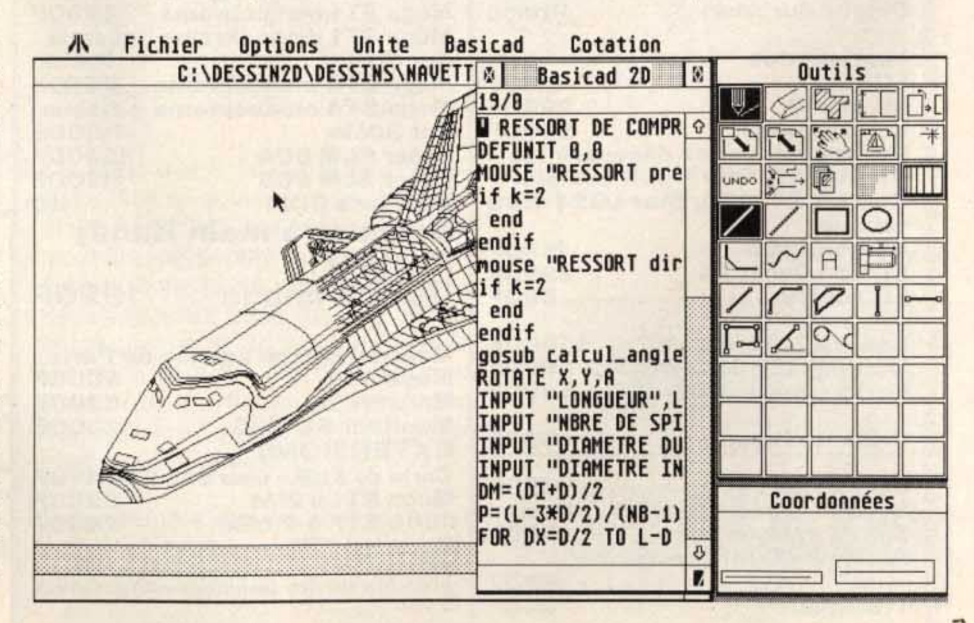
La deuxième zone comprend les outils correspondants à l'option choisie, et la dernière permet d'appliquer une contrainte sur le positionnement des points sélectionnés. Il semble que l'utilisation d'une tablette graphique n'ait pas été prévue, ce qui est regrettable puisque le dessin à la souris présente pas mal d'inconvénients. Une remarque sur la sélection des objets à l'écran : la forme du pointeur de la souris en croix n'est certainement pas la plus adaptée à cette opération. La sélection de toutes ces options/ sous-options étant vite fastidieuse, il est possible de créer très facilement jusqu'à 52 macro-commandes (sauvegardables) qui, par simple pression d'une touche, effectueront tous les choix prédéfinis.

ÉLÉMENTS DE CONSTRUCTION

Il est souvent nécessaire, pour produire un dessin efficacement, de créer des éléments de construction qui seront effacés une fois le dessin terminé. Ces droites seront créées sans contrainte, en parallèle, formant un angle donné (paramétrable ou orthogonal) avec un segment ou prolongeant ce dernier...

CONTRAINTES DE SÉLECTION

Le dessin technique se doit d'être très précis. Lorsqu'on met bout à bout des éléments, il se produit fréquemment un décalage à l'impression, parce que la résolution de l'écran est insuffisante. Ainsi, deux éléments qui semblent se toucher n'ont pas toujours un point commun, mais peuvent être décalés



d'un dixième de millimètre. L'impression sur un périphérique haute résolution, comme un traceur ou une imprimante laser fera immédiatement apparaître un tel décalage. Le Dessin Technique fournit donc tout un ensemble d'outils destiné à pointer des positions avec précision. Il est ainsi possible de pointer l'extrémité d'un segment, son milieu, la tangente à un arc, un point quelconque d'un objet (sans pour autant devoir pointer le curseur avec une précision diabolique), le centre d'un cercle, l'intersection de deux objets... De même, il est possible de demander le tracé d'une parallèle à un segment, à une distance paramétrable.

Un seul regret sur cette boîte à outils : l'impossibilité d'imposer une contrainte à la souris, en l'obligeant à se déplacer sur un axe horizontal ou vertical. On pourra, en pis-aller, se rabattre sur la traditionnelle combinaison clavier remplaçant les déplacements souris (alt + flèches).

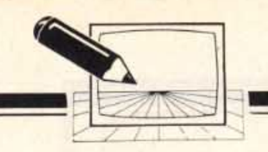
LES OUTILS GÉNÉRAUX

Parmi les options du premier menu, il est possible d'affecter une grille à l'écran, magnétique ou non. Rappelons que le magnétisme consiste à interdire le choix d'un point ne se trouvant pas sur la grille. Il est possible de filtrer l'affichage, c'est-à-dire de le restreindre à certains types d'objets. Le redessin n'est pas automatique, ce qui permet de gagner du temps, car le redessin d'une pièce compliquée prend un temps certain ! De même, l'épaisseur des lignes peut ne pas être affichée, de manière à accélérer le redessin. Il est bien sûr possible de travailler sur plusieurs calques (jusqu'à un maximum de onze), afin de séparer les parties du dessin. Par exemple, dans le cas du dessin d'une pièce formée de plusieurs éléments superposés, il est possible de dessiner chaque élément sur un calque, en rendant invisible les autres, puis à la fin du dessin de rendre tous les calques visibles, de manière à voir l'ensemble des éléments.

La fonction zoom est bien sûr disponible, de deux façons. La première méthode permet de sélectionner la zone de l'écran à agrandir, et le programme calcule le facteur d'agrandissement. Il est possible de basculer à chaque instant entre la vue complète, et la vue agrandie. La deuxième méthode utilise un facteur d'agrandissement défini par l'utilisateur, et permet de se déplacer sur l'écran, en déplaçant le cadre de zone visible sur une vue réduite de la page. Les outils sont tous accessibles en mode zoom. Par contre, si une opération est commencée et que l'on bascule en mode zoom, le programme ne se souvient pas de l'opération en cours.

CHANFREINS ET ARRONDIS

Les pièces techniques comportent souvent un chanfrein, ou un arrondi. Le Dessin Technique permet la création de ces formes, à partir de la sélection de deux segments. Il est



possible de choisir l'angle du chanfrein et sa profondeur, de même que le rayon de l'arrondi. Par contre, l'effet produit n'est pas toujours celui désiré, il est préférable d'effectuer une sauvegarde avant toute création de chanfrein. Il est possible aussi de joindre deux lignes qui ne se touchent pas alors qu'elles le devraient.

HACHURAGE ET COTATION

Le hachurage est souvent la bête noire des dessinateurs, et plus encore celle des développeurs de logiciels de dessin. Le Dessin Technique ne déroge pas à la règle, puisque le hachurage reste une opération délicate, d'où la nécessité d'une sauvegarde préalable. Les hachures peuvent comporter deux types de lignes, d'angle et de distance réglables. La région à hachurer est sélectionnée à l'aide de la souris, et il est possible d'inclure d'autres objets, ou encore d'exclure des objets appartenant à la région sélectionnée; le résultat obtenu est parfois surprenant... Le même problème se pose lors de la cotation des objets, bien que différents types de cotation soient proposés. Il est probable que ces difficultés proviennent de la conception de base du logiciel qui ne gère que des segments et arcs courbes, et ne connaît pas la notion d'objet de type surface : un logiciel 2D qui ne connaît que des objets 1D. Ce problème se retrouve souvent dans le cas des modeleurs 3D qui fonctionnent souvent principalement avec des objets 2D.

L'IMPORT-EXPORT

Le Dessin Technique permet d'importer ou d'exporter des dessins. Les formats reconnus sont en importation AUTOCAD (.DXF), GRAPHTEC (format de traceur Graphtec) et HUMAN (.ASC), pour les logiciels ZZ2D, Dynaccad et ZZ Volume. Les formats d'exportation sont les mêmes que ceux d'importation, auxquels s'ajoutent les formats Métafile (.GEM, si un driver Gdos Metafile a été chargé), et PostScript (à travers l'option d'impression) pour l'export vers des logiciels de PAO comme Publishing Partner Master, PageMaker...

Les limites de communication avec les logiciels de CAO/DAO tiennent aux réserves apportées précédemment sur le fonctionnement du logiciel. En particulier, si l'on veut importer un dessin au format Autocad comportant des hachures, il est nécessaire d'enlever ces hachures dans Autocad pour importer le fichier sans problèmes.

BIBLIOTHÈQUES DE SYMBOLES

Il est possible de créer des bibliothèques de symboles, que l'on peut sauvegarder pour une utilisation ultérieure. Ce sont des objets dessinés que l'on utilise souvent. La notion de pièces est quant à elle plus intéressante. Nous vous l'avions dit plus haut, le Dessin Technique permet la création de programmes en Basic. Une pièce est en réalité un programme que l'on charge en appelant l'option Poser pièce, pour être automatiquement exécutée.

BASICAD : LA PROGRAMMATION

La principale originalité de ce logiciel par rapport à ses concurrents tient dans le langage de programmation intégré : une option se retrouvant habituellement dans des logiciels d'un prix 10 à 20 fois supérieurs ! Basicad est un Basic qui ressemble assez à son grand frère GfA. Les numéros de lignes et les procédures externes sont inexistants, ainsi que l'obligation d'une instruction par ligne qui favorise la création de programmes structurés.

L'Editeur est assez sommaire, mais possède les fonctions de base nécessaires (Copier/Coller...), il est aussi possible de paramétrer l'environnement (unité, type de ligne, couleur). Certaines variables réservées permettent de connaître cet environnement, mais attention, elles ne sont pas protégées, alors évitez de les modifier sous peine de surprises. Les fonctions mathématiques et logiques de base sont présentes, ainsi que la fonction de génération d'un nombre aléatoire.

Les instructions de dessin sont peu nombreuses (droite, cercle, arc, ellipse, boîte), mais suffisantes dans la plupart des cas. Chaque programme peut être exécuté suite à chargement dans l'éditeur, ou en utilisant l'option Poser pièce, laquelle lance alors l'exécution. Les fonctions d'entrée-sortie paramètrent le dessin d'une pièce conçue en Basicad : on questionne par exemple l'utilisateur sur la dimension de la pièce. La souris y est gérée, ce qui permet de demander à l'utilisateur de pointer un objet ou un point de l'écran.Bien entendu, toutes les instructions classiques de déroulement sont implémentées : boucles For Next, While Wend, Do Loop, Exit If... Basicad travaille de surcroît avec des variables réelles en double précision. La programmation en Basicad est très facile d'accès pour qui connaît le Basic, et l'on est vite conduit à ne plus pouvoir s'en passer. Un seul regret : toutes les fonctions du logiciel ne sont pas accessibles directement en Basicad; peut-être dans une prochaine version...

LA DOCUMENTATION

La documentation constitue le point noir de ce logiciel. Elle est plus proche d'un catalogue d'options que d'une véritable documentation, et avec une orthographe douteuse, donne la désagréable impression d'un ouvrage bâclé. Si un exemple de création de dessin est donné et détaillé dans les débuts, il n'est que survolé lorsqu'on aborde les opérations délicates comme le hachurage.

L'IMPRESSION

Le Dessin Technique gère l'impression sur les trois types de périphériques les plus courants : imprimantes matricielles et laser (SLM, PostScript...), et les tables traçantes compatibles HP-GL. Il est possible d'imprimer sur port série, parallèle, ou de créer un fichier sur disque. L'impression est de bonne qualité.

UN OUTIL PÉDAGOGIQUE ?

Le Dessin Technique permet, pour un prix très inférieur à ses concurrents, de s'initier au dessin industriel, et à la programmation CAO. C'est un logiciel d'une grande simplicité d'utilisation, qui ne se fait pas au détriment de la puissance. On pourrait voir là un logiciel d'initiation pour l'enseignement du dessin technique sur micro-ordinateur, pour un prix total (ordinateur et logiciel) très intéressant. Il fonctionne pour le moment en monochrome, avec 1 Mo de mémoire vive. La couleur est prévue pour les versions futures, qui verront probablement une gestion améliorée des traceurs, et souhaitons-le, la possibilité d'utiliser une tablette graphique.

Denis Delbecq

KOMELEC « Grand Public »
Métro REPUBLIQUE
4, rue Yves Toudic
75010 Paris
Tél. : (1) 42 08 63 10
(1) 42 08 54 07
Fax : (1) 42 08 59 05

KOMELEC « Professionnel »
Métro GALLIENI
« Le Carnot »
12, rue Sadi Carnot
93170 Bagnolet
Tél. : (1) 43 63 64 64
Fax : (1) 43 63 77 32

KOMELEC

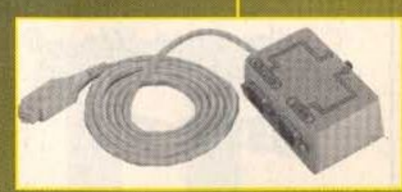
La liaison informatique



Adaptateur Vidéo
CGA PC → Télé
500 F



Commutateur Vidéo
ATARI Multisync
300 F



SWITCHER de Joystick
et de souris
130 F



Sortie audio type RCA
pour ampli ext.

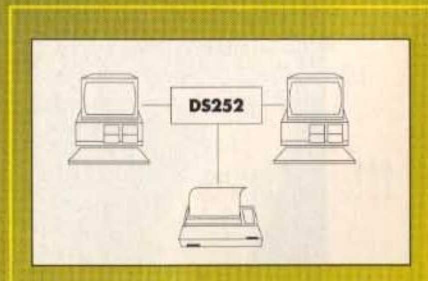
Commutateur Vidéo
mono / coul
200 F



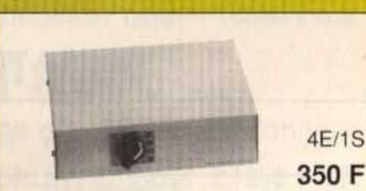
Lecteur ATARI
Switch
300 F

ATELIER DE CABLAGE A VOTRE DISPOSITION

DATA SWITCH



DS 252 *2E/1S*	*210 F*
DS 25 *X2E/2S*	*356 F*



4E/1S
350 F
Câble DATA SWITCH/Micro 25M/25M

Commande mini. : 100 F - Administrations & Sociétés : bon de Cde minimum : 500 FHT. - Catalogue (précisez PC ou ATARI) 30 F remboursé à partir de 200 F d'achat. Port : + 40 F jusqu'à 3 kg. Prix indicatifs. Extrait de notre catalogue connectique :

KOMELEC c'est aussi : Toute la Connectique pour APPLE, PC et AMIGA...

LES PRIX S'ENTENDENT TTC

CABLES INFORMATIQUE SUR ATARI

CABLES IMPRIMANTE

1,80 m	48,00 F
3,00 m	80,00 F
5,00 m	120,00 F
7,00 m	130,00 F

CABLES SÉRIE 25 M/M ou M/F

1,80 m	48,00 F
3,00 m	80,00 F
5,00 m	120,00 F
7,00 m	130,00 F
10,00 m	222,00 F

CABLES RONDS AU METRE

14 Non Blindé	17,00 F
14 Blindé	30,00 F
(Câble en nappe au mètre 0,40 le PTS)	
14 Conducteurs	5,60 F

Câble péritel ATARI	**90 F**
Cordon HARD COPY	**180 F**
Détecteur de sonnerie	120 F
Câble minitel ATARI	95 F
Câble disque dur ATARI DB 19 M/DB 19 M. 0.70 m	120 F
Câble pour connecteur lecteur 5" 1/4	130 F
Câble pour connecteur lecteur 3" 1/2	130 F
Rallonge joystick 1,80 m	70 F
Rallonge joystick 0,30 m	40 F
Rallonge moniteur DIN 13 M/13 F 2 m	130 F
Rallonge lecteur DIN 14 M/14 F 2 m	130 F
Commutateur vidéo en kit	130 F
Câble Null Modem (2m) PC →PC ou PC →ATARI	**100 F**
Câble MIDI Atari 3 m	60 F
Câble MIDI Atari 5 m	72 F

CONNECTEURS

DB09 MALE/FEMELLE	4,00 F
DB15	5,00 F
DB19 "	**8,00 F**
DB23 "	**8,00 F**
DB25 "	6,00 F
DB37 "	12,00 F
DIN ATARI 13 M	20,00 F
DIN ATARI 13 F Ci	20,00 F
DIN ATARI 13 F (Cordon)	30,00 F
DIN ATARI 14 M	20,00 F
DIN ATARI 14 F Ci	20,00 F
DIN ATARI 14 F (Cordon)	30,00 F
Capot DB 9 Vis longues	6,60 F
Capot DB 19 "	**8,50 F**
Capot DB 23 "	**8,50 F**
Capot DB 25 "	6,80 F

MEMOIRES

27256	30 F	41464 - 8	32 F
27C256	32 F	41 1000	65 F
27512	66 F	43256 - 8	75 F
4164 - 10	19 F	44256 - 8	64 F
41256 - 10	19 F	6264	45 F
		68000	90 F

ONE - O - ONE 46 04 88

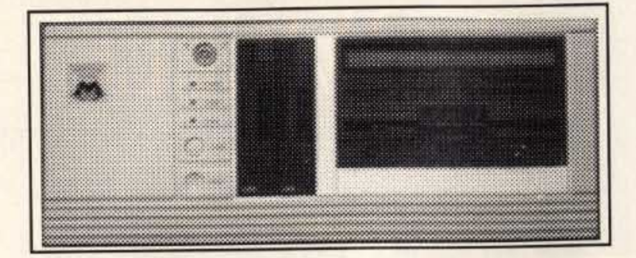
BARETTES SIMM
1 Mo x 8 et 1 Mo x 9 430 F
256 K x 9 250 F

PRIX PAR QUANTITE
NOUS CONSULTER

KOMELEC « Grand Public »
du lundi au samedi de
10h à 12h30 et de 13h30 à 19h

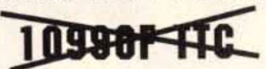
250 F
SOURIS ATARI

LA MEGASTATION



SUPER PROMO!!!

Configuration 1Mo + Disque Dur 20Mo + Lecteur 5 1/4 + Lecteur 3 1/2 + Carte SCSI

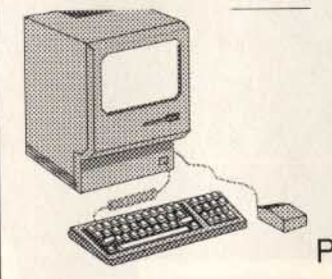


~~10990F TTC~~ **7990F TTC**

PRIX VALABLE POUR TOUTE COMMANDE PASSEE AVANT LE 28/02/91

L'EMULATION

L'ATARI EST UNIVERSEL!





SPECTRE **3490F TTC**

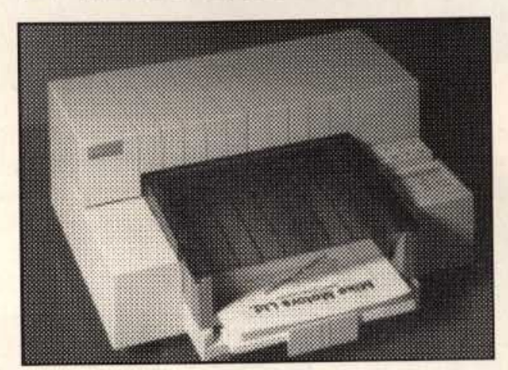
PC SPEED **1290F TTC**

AT SPEED **2690F TTC**

STE, nous consulter

SUPERCHARGER **2990F**

L'IMPRESSION



QUALITE LASER 300dpi

5990F TTC

Extensions Mémoire

Gamme STE

Ext à 1 Mo	390,00
Ext à 1 Mo	1490,00

Gamme STF

Ext à 1 Mo	590,00
Ext à 2,5 Mo	1490,00

Gamme Méga ST

Ext à 2 Mo	1490,00
Ext à 4 Mo	2490,00

ECRANS SUPER !!!

Multisync **3990 F TTC**

Multisync NEC 3D **5490 F TTC**

DISQUES DURS GENIAL !!!

48 Mo 28ms **3990 F TTC**

80 Mo 28 ms **4990 F TTC**

Syquest 44 Mo amovible **5990 F TTC**

52 Mo 17 ms (silencieux) **4990 F TTC**

105 Mo 17 ms (silencieux) **6990 F TTC**

ET TOUJOURS ...

HANDY PARTNER

Le 400dpi,

à moins de 2000 francs.

Complément indispensable

à la PAO, il sauve au format

IMG entre autres.



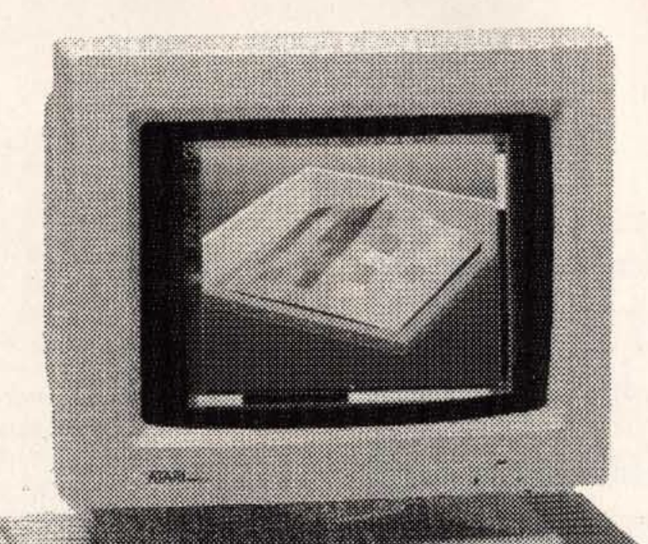
1990F TTC

LIVRE AVEC IMAGE PARTNER!

AVEC ATARI
ON EST TOUS DES PROS !

CENTRE DE COMPETENCE AGREE





LE MICRO DES ANNEES 90

Microprocesseur 68030 - 32 Mhz
Mémoire extensible à 26Mo Disque dur 48Mo
Nombreuses résolutions graphiques jusqu'à 1280x960, Sortie audio Interfaces: //, série synchrone (2), série asynchrone (2), DMA, MIDI
Réseau local, SCSI, Bus VME

Les **cadeaux** de Fevrier !!!

TT 2MO

livré avec **disque dur 80 Mo** et Moniteur **VGA couleur** pour **16950 F H.T.**

TT 8MO

livré avec **disque dur 80 Mo** et Moniteur **multisync** pour **21950 F H.T.**

CONFIGURATION MODULABLE (Consultez-nous)

Ces prix sont valables jusqu'au 15 Mars 91 et annulent les précédents.

Operations dans la limite des stocks disponibles

Certains articles et certains prix peuvent être différents en Belgique !

Le TT peut ne pas être disponible dans certaines boutiques. Renseignez-vous !

MICRO VIDEO

la passion d'un spécialiste, la puissance d'une chaine.

A Paris:
8, rue de Valenciennes 75010 Paris
40.37.92.75 / 40.34.97.80 +
Métro: Gare de l'Est / Gare du Nord.2

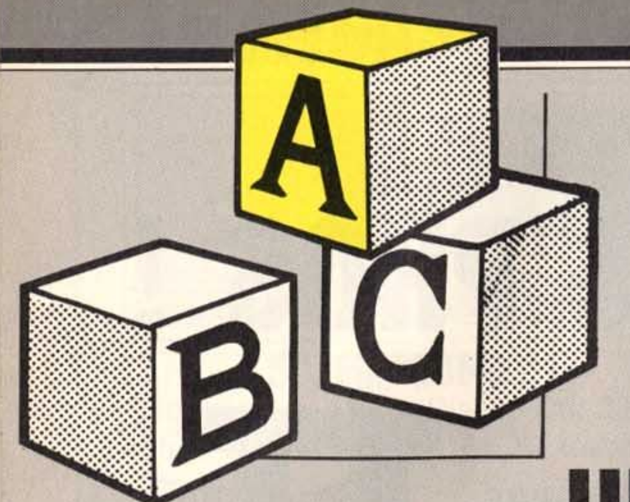
NANTES	6, rue Mazagran	40 69 15 92
PERPIGNAN	8, avenue de Grande Bretagne	68 34 24 40
TOURS	81, rue Michelet	47.05.78.50

En Province:

BORDEAUX	3, cours Alsace et Lorraine	56.44.47.70
DAX	56, Avenue Victor Hugo	58.74.18.63
LYON	11, cours Aristide Briand 69300 Caluire	72.27.14.74

En Belgique:

BRUXELLES	1, rue Dons 1050 Bruxelles	02 / 648 9074
DINANT	21 place communale 5198 ANHEE	082 / 611451



LES ÉDUCATIFS

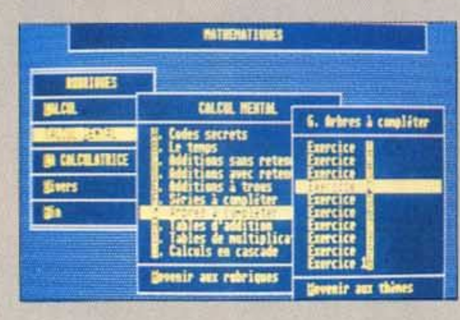
UNE NOUVELLE VAGUE

CALCUL CE1
Moyenne résolution
Éditeur : Hatier logiciel

Ce logiciel fait partie d'une série de dix disquettes, cinq pour le français et autant pour le calcul, qui couvrent le programme de l'école primaire, du cours préparatoire au cours moyen deuxième année. Chaque disquette, affirme l'éditeur, compte cinq cents exercices de difficulté progressive. Nous n'avons pas poussé la conscience professionnelle jusqu'à vérifier cette allégation, ce qui est sûr, c'est qu'il y en a plus que suffisamment. Hatier éditant des livres, il est normal qu'il fasse référence à ses ouvrages, même si leur possession n'est nullement indispensable. Il faut simplement savoir qu'aucun rappel de cours n'est fourni.

Le logiciel est donc un cahier d'exercices, sans doute issu du monde PC, puisque la seule concession au graphisme est la présence de menus en cascade, habituels sur les compatibles. Simple exemple, la calculette ne fonctionne qu'au clavier, et encore selon une disposition des touches différente de celle du pavé numérique du ST.
Les activités sont nombreuses, du calcul aux opérations, via les mesures et la géométrie. Chacune donne accès à dix exercices, sans corrigé particulier, seule la bonne réponse est fournie, ce qui paraît quelque peu sommaire.





BEACH STORE
De la quatrième à la seconde
Basse résolution
Éditeur : Hatier logiciel

Vous voilà à la plage, gérant de la buvette. La météo annonce du beau temps le matin, mais de la pluie l'après-midi. Peu importe, vous avez constitué votre stock en conséquence. Les chalands passent, certains vous font la conversation, d'autres se restaurent. Les heures défilent, et puis, il se fait tard, et c'est l'heure de fermer. Le patron arrive, furibard, vous n'avez pas assez vendu. You're fired.

Cette simulation vise à faire de vous le maître des verbes modaux : Must, May et Can, et de leurs équivalents. Leur subtilité d'emploi n'aura plus de secrets pour vous après quelques journées à la plage. En cas d'erreurs, des messages d'aide (sur quatre niveaux), de plus en plus circonstanciés vous remettront dans le droit chemin. Un mode Apprentissage permet de travailler les différentes variations de sens, comme l'obligation, la capacité, la permission ou la probabilité ou le conseil. Un dictionnaire, disponible en permanence, recense le vocabulaire mis en oeuvre.

Le manuel, fort complet, explique non seulement comment employer le logiciel, mais fournit aussi quelques rappels grammaticaux. Enfin, pour plus de variété, il est possible à un anglophone confirmé de rentrer ses propres questions, et les aides associées. Là encore, la mentalité PC réapparaît, car cette manipulation se fait à l'aide d'une codification fort peu conviviale, même si elle n'est pas compliquée.
En résumé, le mélange jeu/exercices fait encore ses preuves. Et quand on est peu fier de s'être fait éjecter par le boss, on s'applique encore plus à répondre vite et bien.



DENVER présente
JE DÉCOUVRE LES COULEURS
JE DÉCOUVRE LES FORMES
3-8 ans
Basse résolution
Éditeur : Loriciel

Rien de tel que d'inventer un personnage sympathique pour attirer les enfants, et les fidéliser à une série. Denver le dragon est donc né, et son portrait immortalisé par un badge en métal. Second atout, l'ambiance est vivement colorée et les



icônes de bonne taille. En revanche, même si, a priori, le mode d'emploi est destiné aux parents, on souhaiterait qu'il soit écrit en plus gros caractères, car la tranche d'âge visée est celle où l'on apprend à déchiffrer et à lire.

La première disquette est un album de coloriages (14 en tout). Les outils de peinture sont on ne peut plus élémentaires, puisque l'enfant ne dispose que d'un pinceau, qui remplit une forme de la couleur choisie, d'une éponge qui efface tout et d'une gomme qui supprime la dernière teinte appliquée, et ce jusqu'à rendre à l'image son aspect initial. Si certaines nuances sont immuables, il en existe neuf que l'on peut fabriquer grâce à la « machine à couleurs ». Ce qui est un excellent moyen de comprendre comment les trois teintes primaires, le cyan, le magenta et le jaune composent un nuancier infini.

Testé par des enfants, le logiciel est soigné, et les manoeuvres à risques, comme l'effacement, doivent être confirmées. En revanche, on ne trouve aucune procédure de sauvegarde du dessin finalisé. On regrettera aussi l'absence d'un mode de peinture «à pinceau levé », car ici, l'application d'une teinte est toujours parfaite au niveau des contours. L'absence d'une tablette graphique économique (quelques centaines de francs) est toujours pénalisante pour développer des logiciels éducatifs pour les tout petits, et pourtant, ce n'est pas à la souris que l'on apprend à écrire, dessiner et peindre.

La seconde disquette est une variation sur les puzzles. De nombreuses options ajustent le niveau à celui de l'enfant : le nombre de pièces varie de 5 à 80, une grille facilite leur positionnement et le dessin reconsttué





apparaît à la demande. Un système de correction automatique signale les pièces incorrectement placées, et les retire du puzzle. Bien sûr, tous les dessins ne manquent pas de montrer ledit Denver dans ses activités dragonesques et quotidiennes.

Il est impossible de remplacer les images d'origine par celles, mises en couleurs, du logiciel précédent.

ADI
Anglais
5e
Basse résolution
Éditeur : Coktel Vision

Dans un double souci de cohérence et de sympathie pour nos amis bambins, Coktel s'attaque au marché pédagogique à bras-le-corps : il vient de créer la série ADI qui offre un environnement de travail complet, du cours préparatoire à la troisième, concernant le français, les mathématiques, l'anglais et les matières annexes.



ADI, c'est un extraterrestre dessiné en 3D, qui commente le travail de l'élève, et montre divers sentiments qu'il ponctue d'exclamations variées selon la réussite des exercices. Il propose des pauses, et raconte des histoires. Une disquette Environnement, une par classe, donne accès aux disquettes applications.
C'est elle qui propose des évaluations comme des activités de détente (des jeux deviennent peu à peu disponibles, à mesure que l'élève progresse - toujours le vieux mais, ô combien, efficace principe de la carotte et du bâton). L'élève dispose d'outils, un bloc-notes ou une calculette, des documents encyclopédiques, des aides pour rappel du cours et des conseils pour résoudre les exercices.

Le logiciel enregistre les erreurs commises pour activer des révisions ou un retour aux exercices de base. Des tableaux récapitulatifs offrent un suivi informant sur le nombre d'accès, les exercices réussis et les notes. La disquette Application, anglais, mathématiques, français, seront bientôt suivies de disquettes plus spécialisées pour effectuer des tests ou approfondir un sujet. La galette anglaise est une découverte des USA en cinq étapes, chacune proposant des exercices sur les intonations, le vocabulaire ou la grammaire.

Ce tableau montre donc le souci de concrétiser un projet éducatif cohérent, ce qui est tout à fait louable. Corollaire de cela, on imagine que son application au premier cycle de l'enseignement secondaire permet de répartir les coûts fixes de développement.

TRÉSOR DE LA CONJUGAISON
CE2-6e
Éditeur : Micro-C



S'il n'est pas sûr que la conjugaison soit vraiment un trésor convoité par les enfants, il est certain que sa connaissance est un passage obligé vers une écriture correcte. Micro-C propose donc sept mille verbes, qu'il malaxe à tous les temps de l'indicatif, ainsi qu'aux présents du subjonctif, du conditionnel et de l'impératif. Cédant à une mode bien agréable, l'interface simule le relief.
Une première option propose d'afficher la conjugaison d'un verbe donné à un temps donné. Ensuite, des tests ciblés (sur un mode ou un temps) permettent de juger des



connaissances. L'analyse des réponses est intelligente, et, comme deux tentatives sont offertes, elle indique les erreurs par un commentaire éclairé, moins abscons qu'un Faux ou Bon. Enfin, un jeu d'aventure - un parcours dans un labyrinthe à la recherche d'un trésor - est destiné à ceux qui sont assez forts pour faire face à n'importe quel verbe à n'importe quel temps. En effet, le seul moyen de gagner des points de vie est de répondre à des questions. L'ensemble est bien réalisé et agréable à manipuler. Le nombre de verbes et le jeu final garantissent une durée de vie importante. Que demander de plus ?

Laurent Katz

VENTES - RP

Vds Atari 520STF S-F + mon. SM124. TBE: 2200F. Modules mémoire SIMM 256ko x 2: 200F. Tél: 42.00.07.03.

Vds Atari 520 STF 1Mo + imprimante DMP 2160 + mon. couleur + 40 disquettes + livres et revues : 5000F. Tél: 64.01.07.76.

Vds 1040STF (Sept.89) + SM124 + Joy + Souris + disquettes : 3200F à déb. Tél: 46.72.37.50 (ap.19h.).

Vds Mega ST4 + SM124 mono + Spectre GCR + Roms 128 + Disk dur 20Mo + Lecteur ext.D-F Cumana : 9900F. Parfait état. Tél: 43.74.96.01. (Mr Oriol).

Pour apprendre le langage C : Pack "Interpréteur C 2.0" + "Sozobon C" (compilateur). Idéal pour programmer GEM! Prix: 400F seulement. Tél: 99.99.31.98. (soirée).

Vds 520STE 2Mo (2500F) - Écran SC1224 (1500F) - Star LC10 (1000F) - Originaux logiciels : STOS Basic (300F) - STOS Compil (150F) - ZZ-Rough (250F) - Calligrapher Pro (800F) - Diaporama (350F) - Great Courts (150F) - Ferrari Formula One (150F) - Guide Weka + disquette ST/STE (300F). Prix si achat groupé. Philippe au: 45.27.37.38 (dom.), 42.86.20.62 (bur.).

Vds logiciel de simulation de mouvements planétaires et vol interplanétaire sur Atari ST monochrome. 150F. H.DUMITRESCU, 4 rue de la liberté. 78280 GUYANCOURT.

Vds Boîtier Tower pour MegaST faces avant et arrière prédécoupées (comme modèle Micro Vidéo) 1500F. Tél: 69.43.38.79 le w-end. 39.02.78.78 poste 357 la semaine : Alexandre.

Les Petites Annonces sont désormais GRATUITES !
(pour particuliers uniquement)

Pas de chèque, donc, mais obligation d'utiliser le formulaire page suivante, en cochant les cases concernées, et en écrivant en majuscules. Merci !

LES PETITES ANNONCES GRATUITES

Vds Portfolio Atari + interfaces parallèle et série. Sous garantie. 2600F. M.PIRES. Tél: 47.30.43.79.

Vds Spectre GCR complet, neuf très peu servi : 2200F. Tél: 42.38.07.32. (rép.) ou le soir. Demander Yovan.

Vds Atari 1040STF + souris. Lecteur interne D-F 720ko. Excellent état. 2000F. Tél: 47.51.07.92 après 19h.

Urgent cause SM, Vds Amiga 2000 + mon. coul. stéréo 1083S + souris + joyst. + 50 disks divers. Garantie 2 ans. Matériels neufs. 7500F. Tél: 39.47.11.24. Christophe.

Vds 1040STF + mon. coul. CM8801 en parfait état + une vingtaine de logiciels : 3500F à débattre. Vds aussi : Wargame Construction Set (100F), 2 joysticks (80F), Inverseur mono/coul. (100F). André au 42.59.18.01 après 19h.

Vds Digitaliseur couleur haute résolution Pro 8805 Print Technik 2000F. Tél: 47.34.62.99.

Vds Thomson T08 + mon. coul. neuf très peu servi + nbx logiciels jeux et éducatifs + souris + stylo optique et interface. Le tout 4000F. Tél: 43.62.54.69 le soir (HATEM).

Vds simulateur de vol Bomber et F-19 Stealth Fighter (originaux): 150F pièce. Disque dur Megafile 30 (sous garantie): 3000F. Atari 1040ST: 2990F. Xavier au (1) 42.51.08.85.

Vds STF D-F + Péritel + Free-Boot + Disks: 2000F. Lect. Ext 5'1/4 compatible PC (D-F): 700F. Le tout + mon. coul. Philips: 4500F. HP28S + HP48SX. Philippe au 40.34.77.89 (20h).

Vds Atari 520ST + mon.mono. SM124 + imprimante Star NL10 + Joystick + nbx livres et disquettes: 4000F. Tél: 45.67.75.81 (Laurent).

Vds Atari MegaST4 (Déc.90) + Disk dur Megafile 30 + mon. mono. SM124 + Manuels + D7 systèmes + logiciel (Timeworks - PAO) + nbreux logiciels de MAO + Facture du matériel. L'ensemble: 9000F. Tél: 34.82.98.71 (20h).

Vds Mega ST1 nu 2980F. Mon. Coul. SC1425 1890F. Etat neuf: 43.06.38.63 (soir).

Vds MegaST1 + SM124, le tout en parfait état. Prix 4000F + softs donnés. Tél: 47.73.52.86, ts les jours sf lundi, après 20h.

Vds STE 2,5Mo 3600F. STF 2,5Mo 3300F. Disk dur 44Mo SCSI complet 3500F. Extension 2Mo pour STF 1300F. Spectre GCR 2500F. Jacky au 16 (1) 48.49.86.41.

Vds scanner universel Print Technik (digitaliseur d'images + copieur thermique), peu servi 4000F. Gabrielle au 46.65.90.93. Laisser messages ou contact dans la matinée.

Vds 520STF + joystick + souris + jeux + manuel. Parfait état. 2000F. Tél: 60.01.24.00 (après 19h).

Vds MegaST2 + SM124 + lecteur ext. 3"1/2 D-F: 4600F. Vente séparée possible. (78). Tél: 30.93.96.80 (rép.).

Vds 520STF étendu 1Mo + 2 lecteurs 3"1/2 + mon. SC1224 + nbx softs + livres: 3500F à débattre. Tél. Christophe: 43.37.06.84 (semaine), 30.70.89.26 (w-end).

Vds Atari 520STF 1800F. Tél: 45.67.38.74 (Igor, de 18h30 à 21h).

Vds 1040STF, lecteur D-F + très nbx jeux + utilitaires (TTX, PAO, Graphisme): 2800F. Demander Laurent au 42.23.98.98 de 19 à 23h.

Vds Atari 1040 + écran + souris + STBasic et Notices: 3900F. Tél: 41.59.68.37 (J-Luc).

Vds ext. mémoire Amiga 512ko avec horloge et interrupteur: 400F. Tél: 34.11.39.57 (Pascal, après 20h).

Vds 1040STF + SM124 + nbx logiciels 2400F. Tél: 48.83.65.09.

Vds 1040STF mono. + Imprimante Star: 5000F. 1040STE + écran mono.: 4200F. Tél: 42.77.74.56 (Bertrand).

Atari 520STF + Mon. SM124 (NB) + Disk dur SH204 20Mo + Expandeur Roland MT32 (poly 8 voies) + Synthé Yamaha DX27. Le tout en parfait état. 8800F. Marc NADER. Tél: 46.03.30.54. Fax: 47.12.93.01.

Vds MegaST2 + SM124+HD 20Mo + Joystick + Mark Williams C + Minix. L'ensemble: 9500F. Tél: 43.58.12.21 (Bureau, Mr JEN).

Vds Textomat + Datamat + Calcomat2: 450F. Degas Elite 150F. La Bible du ST (av. disquettes) 175F. Programmation Basic 100F (livre + disquettes). Pro Tennis (Great Courts) 150F. Tél: 45.23.17.52.

MegaST4 (TOS 1.4) + SM125: 6000F. En cadeau digitaliseur audio ST Replay4 et collection ST Mag. Jean-Louis PERICARD. Tél: 48.41.77.24.

Vends Synhté Casio CZ1000 TBE 2000F ou échange contre imprimante compatible Atari ST. (Paris). Tél: 40.10.23.13.

Amiga 500 (6/88) + Drive ext. Nec + extension A501 + GfA Basic + 3livres + 50 disks : 3500F. Tél: 39.93.38.70. (ap.20h).

Vds lecteur ext. Cumana 3"1/2 DF 700F, et ST Replay 4.01 soft + hard 400F. Tél: 64.21.14.46. (Laurent après 19h).

VENTE PROVINCE

Vds MegaST1 + mon. mono. + Megafile 30 (séparé 2900F) + Star LC10 + nbx logiciels, disquettes et revues: 8500F le tout. Tél: 98.80.75.41.

Vds Mega ST1 Neuf(1990)- mono + logiciels + souris + boîte de rangement. Sous Garantie. Tél: 41.44.44.38.

Pour ST, Vds imprimante Star LC10 + 5 rubans + cordon, Cyber-Paint, Studio et Sculpt, Chess, Aegis Animator, Dimension3, Rédacteur 3.10 NB et C. Tél: 46.34.18.89 (rép.) - La Rochelle.

Vds originaux sur ST : Explora 2 180F, Crazy Cars 2 100F,Sickworm 100F, Compil. Precious Metal 160F, Music Construction Set 80F. Le tout dans boîte d'origine. Tél: 93.28.00.73 (Jacques).

Vds MegaST2 5600F. Megafile 30 2000F. Spectre GCR + Roms 128k 2500F. Bordeaux. Tél: 56.57.32.47, 56.05.12.53.

Vds + de 400 Démos sur ST tous genres : Digit, SlideShow, Soundtrack, X... Liste contre 1 timbre. THULLIER Stéphane, 9 rue G.Basquin. 59810. LESQUIN.

Vds SuperCharger (v.1,4 - 1Mo) + disk dur Megafile 30. Mr RENARD. Tél: 31.62.27.36 (Lisieux. 14).

Vds MegaST2 + mono SM124 + Disk dur 20Mo + Citizen 120D. Parfait état. Très nbx logiciels (util. + jeux). Prix 10.000F. Loiret. Tél: 38.72.17.15.

Vds Moniteur VGA COul. Sony Trinitron 14" Pitch 0,25/ Digitaliseur vidéo Pro8805 + Caméra + Filtres RVB + Doc (F) + Soft/ SuperCharger 1.44 + DOS 4.01 + 1Mo Ram + Utilitaires/ Lecteur 5,25" 40/80 pistes ST-PC/ PC Ditto 3.96 + DOS/ Prix à débattre. Tél: 35.70.48.88.

Pianos Stéréo MIDI Casio MT260 - 210 sounds + kit batterie avec alim. 220v.: 1500F (valeur 2200F). Recherche MidiMix. PAVAN Pierre. 2 rue du Lac. 25660 SAONE.

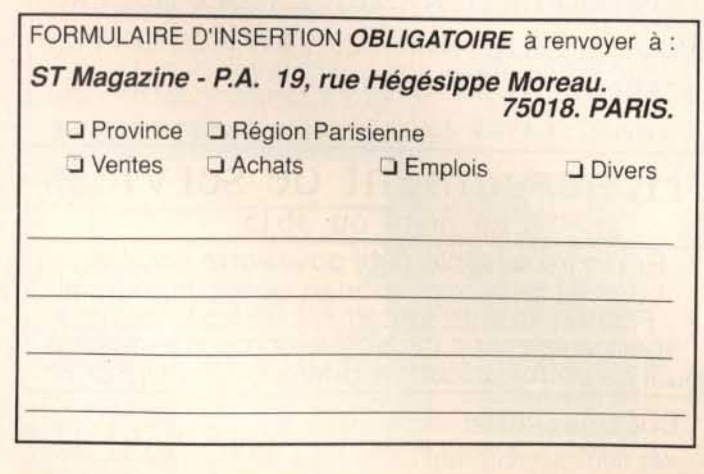
Urgent! Vds C64 + lect. disquette 5'1/4 + 2 joysticks + nombreux jeux et utilitaires : 2000F, Tél: 88.62.27.10. Bas-Rhin (Éric après 18h).

Amiga 500 garanti + Joystick + souris + nbx logiciels (création graphique et simulation), le tout 3200F. Tél: 30.82.05.12. après 20h. Mr Giraud.

Vds extension mémoire pour monter 520 STE en 1040 (2 modules SIMM de 256ko) 300F + cherche contacts région Trégor. Tél: 96.47.04.54 le soir (Yann).

Vds Moniteur coul. Europhon Pal/Secam (Péritel) 1600F. Message au 64.45.71.24.

Vds lecteur externe Cumana 3"1/2 pour Atari 700F + port. Tél: 68.78.79.00. HdR.


FORMULAIRE D'INSERTION ***OBLIGATOIRE*** à renvoyer à :

ST Magazine - P.A. 19, rue Hégésippe Moreau. 75018. PARIS.

❑ Province ❑ Région Parisienne

❑ Ventes ❑ Achats ❑ Emplois ❑ Divers

Vds moniteur NEC 2A (multisync) 800x600: 3900F (neuf, jamais utilisé). 520STF D-F new Roms, gonflé 1Mo + SC1425: 4500F. Sélecteur de face 50F. Hard copieur 250F. Tél: 50.54.51.66 HR.

Cause achat laser, vds Citizen 24 aig. HQP40 + Kit coul. Neuve, sous garantie, Val 3500F vendue 2200F. Originaux For Red OCtober 100F, 1ST Word Plus 2.02 300F. IMBERT 2 rue Berlioz 69680 CHASSIEU.

Vds pour ST: Popuklous (100F), Robocop (100F), Barbarian II (100F), Dragon Ninja, Black Tiger et Thunder Blade à 70F. Tél: 74.77.44.27 (Olivier, W-end seulement).

Vds MegaST1 + Guide Weka (disk) + IMG Scan: 4200F. 520ST 1Mo: 1800F. Impr. coul. jet d'encre: 3800F. Casio PB1000: 1000F (ext.mémoire). Tél: 68.26.33.26. ou 68.26.30.71.

Vds SM124 700F. Sons Synthé studio (M1, M3R, V50, D70, Proteus, K1, K5, DX7IIFD, TX81Z, S550, W30. Logiciels originaux V50, YS200, K4. Tél: 61.80.87.82.

DIVERS

HP48SX : Vds HP48SX neuve, cause double emploi : 2400F. Denis LAFONT, 57 rue de Tassigny 76130 Mont St Aignan. Tél: 35.75.19.43 (rép.).

Graphistes ? Musiciens ? Codeurs? Vous habitez Evry (91) ? Vous voulez faire partie d'un groupe ? Tél. à Arnaud, au 64.97.96.11.

Étudiants ou non, exposés ou compte-rendus, je mets en page pour vous (Calamus, PPM, Timeworks...). Petits prix. Contactez-moi. RACCA Damien. Bois de la Pierre. 31390 CARBONNE. Tél: 61.87.91.39.

Recherche contacts Atari ST pour échanges Jeux, Domaines Publics, Utilitaires. Sérieux et durable. RATA Olivier, 41 rue des Burgues. 11590 Salle les d'Aude.

Existe-t-il un club Atari sur Région Rennes ? Écrire à A. GERARD 9 Allée des Bergeronnettes. 35340 LIFFRE.

Vous avez un ST(E) ? Le CSC Club édite une disquette d'information et de domaines publics. Envoyer une enveloppe timbrée pour plus de renseignements à: RENARD Olivier. 8 Hameau du vieux puits, CH Garibondy, 06110. LE CANNET.

Graphiste (75) cherche programmeur efficace (Amiga + autres micros) pour créer Jeux - Fantasy: Story, Scenar, 1/3 Graphismes faits. Tél: 45.79.88.12 (Jacques).

Recherche contacts avec Prog., Graphiste et Mus. pour échanger trucs, astuces, bidouilles... sur amiga. Et surtout programmes sur AMOS. ROBA Alexandre 53 rue Grande. 7050 Herchies (Belgique).

Recherche contacts sur ST dans le 29 pour création de démos ou autres. Appelez au 98.07.88.67. Patrice, le w-end.

Échange logiciels divers. Tél: 61.41.38.27.

Cherche correspondant en vue d'échange de programmes sur Atari 520STF, de préf. en Bretagne (Morbihan). Écrire à KERLAUD José 8 rue du Lévenant 56400 AURAY. Tél: 97.24.06.52 (w-end).

Tunisie : Cherche contacts ts pays, échanges programmes, idées, correspondances pour Atari ST. Daoud Mahrez 1 rue Nasser Gargouri 3054 Merkez Sebaï. Tunisie.

1040 STE recherche contacts sérieux Musique et Utilitaires. Patrick RAYNAL, Domaine de Gondières 58000 SAINT ÉLOI.

Propose une méthode de cours d'assembleur ST pour débutants par correspondance (cours particuliers possibles sur Paris). Demandez une doc à Cedric JAVAULT, 82 rue de l'Université 75007 PARIS. Tél: 45.44.21.05.

EMPLOI

Important revendeur parisien recherche technicien SAV sur Atari, à temps partiel. Prière contacter Mr SEYOT au 43.38.96.31.

TELEGROUPE recherche des passionnés de programmation connaissant bien le basic GfA, intéressés par le minitel. Autodidactes bienvenus. POste stable. Salaire motivant. Lieu de travail : Paris-Châtelet. Appelez Mme Farjaudon au (1) 40.26.33.33 ou envoyez votre CV au 8 pl. Ste Opportune 75001. PARIS.

ACHATS

Achète DBMan IV ou IV avec Runtime valable. Tél: 88.51.16.35. (Alsace, ap.20h). Pour - 600F.

NOUVEAUX ACCESSOIRES

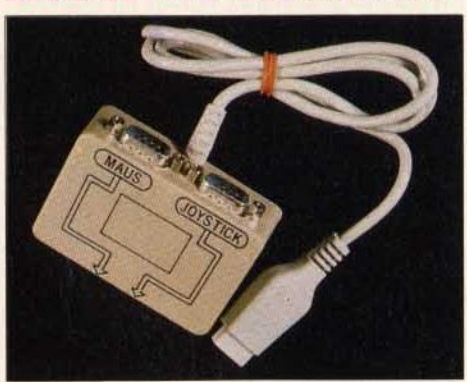
Cette nouvelle année voit l'arrivée de nouveaux petits accessoires chez Komelec, société déjà réputée sur ce créneau. Il s'agit de petits boîtiers, de câbles, de connectique générale, etc. Aujourd'hui, nous avons reçu un switcher pour moniteur multisync ainsi qu'un switcher souris / joystick.

LE SWITCHER MULTISYNC



Il s'agit d'une petite boîte grise, s'intercalant entre l'unité centrale (prise moniteur) et le sus-dit moniteur multisynchrone (si vous avez la chance d'en posséder un). Ce dernier devra comporter obligatoirement une prise DB9 (type joystick) femelle. L'intérêt de ce genre de boîtier réside dans le gain de place sur votre bureau et dans la connectique simplifiée : un seul écran, un seul câble, une seule boîte ! La boîte comporte de plus un bouton poussoir servant à passer de résolutions couleur (basse ou moyenne) à la monochrome, d'un seul coup d'un seul, et présente deux RCA femelles pour les voies audio In et Out (mono). Ce boîtier, pour 300 F TTC, rendra sûrement de fiers services à ceux qui ne maîtrisent pas le fer à souder, les autres ayant sûrement procédé à la "bidouille" proposée dans notre numéro 40.

LE COMMUTATEUR SOURIS/JOYSTICK



Bien qu'aujourd'hui, la plupart des jeux utilisent le port joystick 1, et ne nécessitent donc pas de débrancher la souris, il se trouve encore des situations où cette manipulation est indispensable. Les jeux à deux adversaires en présentent un exemple, et entraînent une fastidieuse préparation. Si vous vous énervez à chaque fois que vous remplacez la souris par votre joystick, ce petit accessoire est pour vous...

Ce commutateur permet, après y avoir connecté à la fois la souris et le joystick, de les sélectionner avec un simple interrupteur. Rien de réellement novateur à cela, mais puisqu'un nouveau modèle apparaît sur le marché, qu'il est de bonne qualité, et d'un prix honnête (130 F TTC), pourquoi ne pas en parler ?



DÉMOS :

OSS, CAPITALE DE L'EUROPE

Du 21 au 24 décembre s'est tenue à Oss, en Hollande, la meilleure "coding party" jamais organisée sur Atari ST. Préparée par les rédacteurs du magazine sur disquette "ST News", elle a réuni les plus grands groupes d'Europe, que nous vous invitons à mieux connaître ce mois-ci et le mois prochain, à travers cette extraordinaire convention.

Mais tout d'abord : qu'est-ce qu'une "coding party" ? Il s'agit simplement d'une réunion entre développeurs de démos ou de jeux, durant laquelle on se montre ses travaux respectifs, on échange ses points de vue et ses productions, on participe à des activités organisées sur place... Il faut souligner que les activités illégales sont totalement bannies de ce type de convention.

"STNICCC"

L'aventure débute pendant l'été 1990 lorsque nous recevons une invitation pour la "ST News International Christmas Computer Conference", qui doit se tenir à Noël à Oss, Pays-Bas. Cette "conférence" doit réunir, pendant presque quatre jours, les principaux groupes européens de démos, ainsi que des professionnels de la micro. Plus on s'approche de Noël, plus la liste des groupes invités grossit, si bien qu'au jour J, tous les grands groupes étaient présents, sauf Inner Circle et Phalanx. Thalion Software (l'équipe au complet) et Tos Magazin (Martin Backschat et Armin Hierstetter) s'étaient également déplacés d'Allemagne pour accompagner l'événement. Et bien entendu, Stefan Posthuma ("Digital Insanity") et Richard Karsmakers (l'auteur des "VDU", travaillant aussi chez Thalion), les deux organisateurs et auteurs du magazine sur disquette "ST News", étaient sur place. Au total, on dépassait amplement les 100 personnes ! La fête pouvait commencer...

"DÉMOS COMPÉTITIONS"

Les concours ST News ont constitué un des temps forts de la convention. Ils se sont divisés en trois catégories : un concours de démos ne pouvant excéder 3.5 ko (compressés ou non), un concours de graphismes et un concours de démos normales (proposé en dernière minute par quelques groupes). Le jury était constitué par Thalion, ST News et TOS Magazin. Les prix décernés étaient des T-shirts, K7 vidéo ou disquettes ST News de la convention, des jeux Thalion Software, des abonnements ou des calendriers TOS Magazin, etc.

Le premier concours a été remporté par DELTA FORCE avec une adaptation du jeu "Archon" en 3.5 Ko. Les deuxième et troisième prix ont été remportés respectivement par DYNAMIC DUO et les OVERLANDERS. THORSTEN (Gogo de TCB, déjà graphiste de Spherical) est l'auteur du meilleur dessin du concours graphique, devant RED (Omega) et SPAZ (Lost Boys). Le prix de la meilleure démo "normale" fut attribué aux LOST BOYS, suivis de NEXT et de DYNAMIC DUO.



L'écran de Manikin et Oxygene.

Ces concours furent une bonne initiative des organisateurs. On peut seulement regretter que peu de groupes y aient participé (à peine le quart). Quoi qu'il en soit, les réalisations qui ont été primées devraient être disponibles sur la disquette ST News "Special Convention" à partir de ce mois-ci. Ce forum était également l'occasion idéale pour sortir et diffuser des démos. C'est ainsi que trois mégadémos sont sorties à la STNICCC.

THE EUROPEAN DEMOS

Elles ont été finies sur place, les Overlanders s'étant enfermés dans une salle, tous ensembles et avec MCoder, pour travailler d'arrache-pied à la finition de cette superproduction. Ce fut chose faite au milieu du week-end, et on peut dire que personne n'a été déçu. Cette mégadémo, occupant deux disquettes

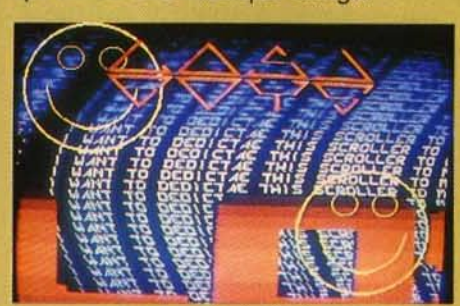
double-face, constituera désormais un poids lourd, un élément incontournable du monde des démos sur ST.

Nous l'avions déjà dévoilée le mois dernier en vous présentant trois écrans, mais il reste tout de même des détails à ajouter. D'abord, le Main Menu. Il a été modifié sur un vieux ST appartenant à un autre groupe français bien connu (et réputé pour ses chaudes critiques envers les overscans qui ne passent pas sur ses ST), mais dont nous tairons le nom. Ceci fait que désormais, le Main Menu marche sur tous les ST : en voilà une bonne nouvelle ! En revanche, faute de temps, les autres écrans en overscan n'ont pas pu être modifiés. Notons également une inflation galopante et tous azimuts de "megascrollers", qui ne constituent assurément pas la plus belle façon de mettre en valeur la technique du syncscroller. Enfin, félicitations à Mr. Bee pour sa démo en "loader", qui est la meilleure à ce jour.

THE HACKABOUND DEMO

Programmée par Total Vision Inc. (The German Alliance), elle est sortie peu avant la convention, mais n'a vraiment été diffusée que là-bas. Il s'agit d'une "convention démo", avec un invité pour l'occasion : les Lost Boys, en la personne d'Oxygene, le membre allemand.

Commençons par leur écran justement... Nous l'aimons bien, parce qu'il a de la classe (quelque chose qui ne s'explique pas, mais qui se constate !), et surtout parce que la musique Soundtracker en quatre voies, tirée de la Yum Yum Demo par Dual Crew sur Amiga, est de bonne qualité (musicale comme acoustique). Il est intéressant de noter que c'est le premier écran des Lost Boys qui utilise une musique Amiga.



L'écran des Lost Boys

Notons également une assez belle page de présentation : un gros logo TVI rebondit sur un fond de plusieurs petits scrolls, avec une musique Soundtracker. Signalons que ceci se passe en overscan, et que d'après ce que l'on nous a dit, la routine utilisée est celle du TCB Tracker. Il y a quelquefois des problèmes de synchronisation sur les vieux ST (quand ils chauffent, on remarque les mêmes problèmes que la démo "Death of the Left Border" de TNT-Crew), mais il suffit de booter avec un ST bien froid pour régler le problème !

THE SYNTAX TERROR DEMO

Nous vous l'avions annoncée il y a déjà deux mois, et l'attente n'aura pas été vaine : voici la Syntax Terror Demo, par Delta Force, accompagné par d'autres groupes.

Commençons donc par l'écran de TEX : il a bien sûr la classe de toute production de ce groupe. Il est en fullscreen, avec syncscroll vertical. La routine de full semble assez instable, puisqu'elle ne marche qu'une fois sur deux, sur plusieurs types de ST. La musique de Mad Max (inédite) est sublime, elle est adaptée de la musique de euh !... d'un film. L'écran est en fait constitué de plusieurs parties que l'on découvre en se déplaçant verticalement. Et c'est là que se situent l'originalité et l'intérêt technique de la démo. On se demande si Daryl gère l'ensemble des animations en une fois (en faisant tenir le tout en une VBL), ou s'il ne gère que ce qui est à l'écran à l'instant présent (ce qui libère du temps machine, mais est plus complexe à gérer). A priori, on serait tenté de penser à la première solution, car il suffit de descendre vers le bas de l'écran pendant quelques secondes et de remonter ensuite, pour s'apercevoir que le scrolltext supérieur ne s'est pas arrêté. Mais connaissant Daryl, on peut dire sans trop s'avancer qu'il a dû avoir recours à des ruses de Sioux... Quoi qu'il en soit, un écran au top niveau par TEX. Pouvait-on en douter ?



L'écran de TEX

Autres intérêts de la Syntax Terror Demo : deux jeux de réflexion, donc passionnants. Leurs noms : "Match It !" et "Soko Ban". Nous n'allons pas détailler leurs règles, mais simplement signaler qu'ils sont programmés de façon rigoureuse (les meilleurs scores sont sauvegardés, par exemple), dotés de graphismes agréables et qu'ils sont dangereux : il est très difficile de s'arrêter, une fois que l'on a commencé ! En fait, ils valent bien mieux que bon nombre de logiciels de jeux disponibles dans le commerce...

Level 16 (le groupe pionnier en matière de fullscreens), signe son retour dans les démos, et joue encore une fois la carte de l'originalité et de l'innovation. Titre de son écran : Multicolor Sound Display. Avec une synchronisation de couleurs, qui varie en fonction de la musique jouée, Ilja donne de superbes effets de couleurs sur tout l'écran. Tout est paramétrable, selon les voeux de l'utilisateur. Superbe. Enfin, pour la petite histoire, notons qu'Ilja ne programme pas de jeux sur le ST, comme ses camarades de The Union ; il se contente de programmer des démos ("just for fun") : il est déjà, dans sa vie professionnelle, programmeur. Et comme il le dit lui-même : " Un fullscreen dans ma vie, ça me suffit !" A méditer... (N.B. : ces propos ont été recueillis lors de l'interview de TEX).

Un cocktail détonnant de sinusoïdes et de rasters, voilà ce que nous propose Delta Force avec la démo "Colorshock III". Après TNT-Crew et TCB, c'est au tour de la force Delta de nous montrer leur choc de couleurs. Et le résultat est superbe : on se trouve face à des animations dignes des démos au Copper sur Amiga. Il est clair que le groupe allemand maîtrise bien la programmation du MFP.

On en reste là pour la Syntax Terror Demo, sinon nous serions bien partis



pour vous parler de tous les autres écrans ! Sachez cependant qu'il s'agit, mine de rien, de la démo en une disquette disposant du plus grand nombre d'écrans (25 au total), qu'aucun d'entre eux n'est mauvais, et que cela fait bigrement plaisir de voir des groupes que l'on croyait disparus se remettre au boulot, au moins pour cette occasion... Alors faites-vous plaisir avec cette démo un brin nostalgique, mais aussi de très haut niveau !



Le "Greetings Screen"

LES PREVIEWS

ULM MEGADEMO : "The Dark Side of the Spoon". Nous l'avons vue dans sa totalité en Hollande, où le groupe luxembourgeois était supposé la finir, et elle est maintenant sortie depuis quelques heures (rajoutez un mois pour vous qui lisez ces lignes !). C'est la nouvelle référence sur ST, encore que de nos jours, les meilleures démos se tiennent toutes dans un mouchoir de poche. Avant d'en parler plus longuement le mois prochain, retenez simplement que le Main Menu est en fullscreen, avec musique soundtracker quatre voies (superbe), et syncscroll multidirectionnel (et non pas simplement haut-bas ou gauche-droite !). "Test complet le mois prochain !"...

SWEDISH NEW YEAR DEMO III : Mais oui, les Suédois ont bien prévu d'en sortir une troisième. On pourra compter sur la participation de Sync, Omega et... Silents. En effet, il semble que TCB soit absent cette année. En échange on aura droit, comme vous l'avez lu, à Silents, le département ST du célèbre groupe Amiga. Et leur nom n'est pas usurpé, car leur écran, que nous avons vu en Hollande, est d'excellente qualité. Aux dernières nouvelles en provenance de Suède, Omega n'était pas satisfait de son écran, ce qui reportait la date de sortie de la démo. Vont-ils devoir également changer le nom pour cause de retard ?

D'AUTRES PREVIEWS !

Beaucoup d'anciens créateurs de démos se sont lancés dans la programmation de jeux. Le groupe allemand TEX en est bien sûr le meilleur exemple, mais il y en a d'autres...

Omega travaille en ce moment sur un shoot'em up nommé Futuria, et qui est pratiquement fini. Nous avons pu tester une préversion en Hollande, et pouvons vous dire que le jeu est sensationnel ! Il est superbe aussi bien graphiquement que techniquement : certains monstres de fin de niveau mesurent en effet deux écrans de long, soit plus que n'importe quel jeu du genre sorti jusqu'ici. De plus, on peut y jouer à deux simultanément. D'ailleurs vous allez pouvoir vous en apercevoir par vous-même, puisque dans la Swedish New Year Demo III, vous trouverez un niveau entièrement jouable de Futuria ! Alors, heureux ?



Une preview de Futuria

Manikin des Lost Boys a fini un jeu de plates-formes, particulièrement jouable et entraînant (sans doute le meilleur du genre), surtout lorsqu'on choisit le mode à deux joueurs, car là on a la possibilité de s'entraider dans les passages difficiles, ou au contraire, de se combattre ! Le jeu s'appelle "A Prehistoric Tale", et sortira très prochainement. Ne le ratez pas, car comme vous avez pu le lire dans l'interview que le groupe anglais nous a accordée, il n'y aura probablement plus de jeux par les Lost Boys du fait du temps de développement que cela implique.

Nick de TCB a fini le jeu que l'on pourrait appeler "Seven Gates of Jambala 2" puisqu'il s'agit de sa suite, mais qui a été baptisé "Enchanted Lands". Sachez que c'est le plus beau jeu qu'il nous ait été donné de voir. On bouge son personnage dans un écran de 320x200, en scrolling multidirectionnel et en une VBL : du jamais vu dans le domaine des jeux ! Et le reste est à la hauteur, notamment la beauté des graphismes et la page de présentation à la Shadow of the Beast qui fera date. Voilà en tout cas de quoi faire tousser du côté de pas mal d'éditeurs de jeux...

Pour ce qui est des utilitaires, il y avait bien entendu Neochrome Master, dont nous parlons dans l'interview de Delta Force, qui offre à l'utilisateur de nombreuses possibilités que la version normale n'avait pas, comme un menu dans l'overscan du bas (ce qui permet de dessiner en plein écran avec la loupe), des formats de sauvegarde qui comblent les programmeurs, des routines de chargement et de sauvegarde TRES rapides, etc.

Enfin, Sync nous a présenté une version quasi définitive de son soundtracker : Audio Sculpture. Et cela dépasse de loin tout ce qui a été fait jusqu'à présent sur ST en matière de soundtracker, même l'excellent "Noisetracker" de Doctor Satan (disponible dans le Domaine Public). Audio Sculpture n'a rien à envier aux soundtrackers de l'Amiga, ce qui veut tout dire ! Redhead (SPreAdhead pour les intimes), son programmeur suédois, vient de nous annoncer la sortie de ce logiciel pour le Cebit'91, ce printemps à Hanovre.

Et comme nous sommes bons princes, nous vous en reparlerons le mois prochain, lors duquel nous terminerons ce voyage à Oss, avec les plus grands groupes suédois, interviewés et photographiés pour vous !

Pour finir, nous aimerions faire part de nos remerciements spéciaux à Armin Hierstetter et Martin Backschat de Tos Magazin, Stefan Posthuma et Richard Karsmakers de ST News et Thalion Software, et Guillaume le "mauve" des Overlanders.

Klaus Berg
Patrick Raynaud

DÉMOS : LES INTERVIEWS

Entretien avec les Lost Boys : Manikin (Tim), Spaz (Dave), Sammy Joe (Michael), Digital Insanity (Stefan) et Oxygene (Fabian).

ST Mag : Des nouvelles de votre prochaine Mega Démo ?

Manikin : On ne peut pas vraiment parler de "nouvelles" : on doit d'abord la commencer ! Mais elle devrait vraisemblablement être finie vers Pâques.

ST M. Voire plus tard...

Man. Oui, probablement ! (rires) Nous cherchons des idées originales. La démo devrait comporter huit écrans environ, et elle sera différente...

ST M. A quel niveau ?

Man. Pas forcément au niveau des écrans eux-mêmes. Mais nous avons un concept particulier pour l'ensemble de la démo. D'une certaine manière, nous allons la considérer comme un film.
Digital Insanity : Les gens sont fatigués de voir toujours le même style de démo, uniquement axé sur la technique. Nous voulons faire quelque chose d'original et de drôle. Définir un nouveau concept.
Man. Sans pour autant négliger la technique, bien sûr.

ST M. Et pour ce qui est des jeux ?

Man. Nous avons fait un jeu : "A Prehistoric Tale", chez Thalion Software. Et c'est probablement le dernier, car la finition est infernale ! J'avais déjà commencé un jeu il y a longtemps, mais à cause de mes études à l'Université, je ne l'ai jamais fini. "A Prehistoric Tale" sera sans doute le seul et unique jeu produit par les Lost Boys.

ST M. Combien de temps avez-vous passé dessus ?

Man. Environ trois mois... Quatre, même : tout l'été en Allemagne chez Thalion, puis à la maison en Angleterre pour le finir.

ST M. Cet été, vous avez aussi pris deux nouveaux programmeurs : Fabian et Stefan, qui ne sont pas anglais, mais respectivement allemand et néerlandais. Pourquoi eux ?

Man. Stefan, nous le connaissions depuis longtemps, et nous lui avions déjà proposé d'intégrer le groupe. Mais au début, il ne voulait pas, car il aurait été trop loin de nous, puisqu'il habite en Hollande et que nous sommes en Angleterre. Nous avons rencontré Fabian cet été chez Thalion et lui avons proposé de se joindre à nous, parce que outre le fait qu'il est très bon, c'était dur de continuer à produire avec un seul programmeur. Et puisque Fabian, qui est allemand, a accepté, Stefan s'est décidé à son tour !
D.I. Oui, je trouve cette optique "européenne" très intéressante.

ST M. Et votre "Maggie Disk" ? Comment t'est venue cette idée, Michael ?

Sammy Joe : ST News s'est arrêté. Je me suis dit qu'on pourrait faire quelque chose pour le remplacer, et donc j'ai commencé un "Maggie Disk", d'abord pour le "fun". Puis j'ai pensé que ça valait la peine de continuer.

ST M. Cela demande beaucoup de travail, non ?

S.J. Pas tant que cela, car beaucoup de gens et de groupes du monde entier apportent leurs articles. Mais c'est sûr, cela nécessite une bonne organisation.

ST M. A quel rythme comptes-tu faire paraître les numéros de "Maggie" ?

S.J. A celui qui nous arrange le mieux ! Non, disons qu'un délai d'un mois et demi entre deux numéros semble raisonnable. Un mois, c'est trop court.

ST M. L'informatique est-elle une distraction, ou bien une activité à laquelle vous aimeriez consacrer votre vie ?

Man. Notre avenir, c'est probablement l'informatique, mais pas celle des jeux pour micro.

ST M. Des applications professionnelles, donc ?

D.I. Oui, d'ailleurs je travaille déjà dans une société informatique.
Man. Et c'est également mon ambition.

ST M. Quel avenir pour les Lost Boys sur ST, à long terme ?

Man. On verra bien, après la Mega Démo...

* * *

Entretien avec Stefan Posthuma et Richard Karsmakers, les organisateurs de la STNICCC.

ST M. Quand vous est venue l'idée d'organiser cette convention ?

S.P. : Cet été, chez Thalion Software. TLB et TCB voulaient se rencontrer dans le cadre d'une "coding party", avec quelques autres groupes. 50 personnes ont été prévues à l'origine, et j'ai pu obtenir les locaux de l'entreprise dans laquelle je travaille. Mais le nombre d'invités supplémentaires n'a fait ensuite qu'accroître. Jamais une convention de ce genre n'a réuni autant de monde, je crois. Il y en a même trop.

ST M. Quel est le but de cette réunion ?

S.P. : Que tous les bons groupes de tous les pays puissent se rencontrer.
R.K. : Il y a ici de nombreuses personnes qui sont très avancées en



"3D", d'autres en soundtracking, etc. Tous pourront s'échanger des informations et avoir ainsi progressé à la fin du meeting.

ST M. Les meilleurs groupes d'Europe sont venus. Seul Inner Circle n'était pas là. Pourquoi cette absence ?

S.P. : Il y a eu un malentendu entre les Lost Boys et Inner Circle. Ces derniers ont accusé les premiers, à tort, d'avoir copié leur routine de "3D". Or, chacun des deux groupes avait optimisé de manière assez semblable (car les optimisations étaient évidentes) une routine publiée dans le magazine anglais "ST World" ; et ce, en gardant les labels d'origine. Inner Circle a ensuite cru que la routine des Lost Boys dans la démo "Life's a Bitch" était dérivée de la sienne, ce qui bien entendu, est totalement faux.

ST M. Avez-vous invité des personnes qui n'ont pas pu venir ?

R.K. : Nous avons invité plusieurs programmeurs "officiels" : Jeff Minter, Steve Bak, Peter Johnson, John Philips, les Bitmap Brothers, Assembly Line... Mais ils n'ont pas pu venir pour des raisons diverses. Je crois qu'ils n'étaient pas réellement intéressés. A mon avis, le seul qui a vraiment regretté de ne pas pouvoir venir est Jeff Minter. En revanche, il nous a envoyé un vrai jeu pour le concours de démos en 3.5 ko ! D'ailleurs, Michael Bittner et Erik Simon (Thalion) sont également en train de faire un jeu pour ce concours : un vrai "shoot'em up" en 3.5 ko, avec plusieurs niveaux ! Ils ont utilisé un système très ingénieux. Je crois que Michael est simplement le meilleur programmeur du continent ! Delta Force a également fait un jeu : Archon.

* * *

DELTA FORCE
Cette interview de Delta Force, avec New Mode, Chaos Inc. et Slime, a été réalisée en anglais et en allemand. Elle a eu lieu dans l'escalier menant au deuxième étage du "Fairy Tale Castle", le samedi 22 décembre 1990...

ST M. Qu'a fait Delta Force depuis la Union Demo ?

New Mode : Rien, vraiment rien (rire général).

ST M. Vous avez bien réalisé un jeu, non ?

Chaos Inc. Oui, j'ai fait un jeu de rôle qui est déjà sorti dans le commerce (NDLR : A la demande de Chaos Inc. nous n'avons pas cité le nom de ce jeu).

ST M. Avez-vous d'autres projets de jeux ?

N.M. Slime et moi travaillons dessus : ce sera un jeu de stratégie.

C. I. Je travaille maintenant sur la suite de mon premier jeu. Cette fois-ci, je programmerai probablement aussi la version Amiga, ce que je n'avais pas fait la première fois.

ST M. Venons-en à la Syntax Terror Demo. Combien de temps vous a-t-il fallu pour la programmer ?

N. M. Tout a commencé à la fin de 1988. 2 écrans et un main menu étaient prêts, et ensuite nous avons décidé d'être paresseux pendant un an et demi ! Mais depuis le mois de juin de cette année, j'ai passé une bonne partie de mon temps dessus.

C. I. Il faut quand même préciser que New Mode a fait son service militaire jusqu'en juin cette année, et que ce n'est que depuis, qu'il a vraiment eu du temps à consacrer à cette démo.

ST M. Quels commentaires sur la Syntax Terror Demo aimeriez-vous faire ?

N. M. Elle est ennuyeuse !
C. I. Disons qu'elle est sortie trop tard ; elle aurait été excellente si elle était sortie il y a six mois ou un an. Aujourd'hui elle est très bien, mais ce n'est pas LA démo. Il n'y a pas d'écrans sensationnels, seulement de bons écrans et de bonnes idées. Par contre, et c'est là une nouveauté, la démo comporte deux jeux. Ils sont, à mon avis, tous les deux meilleurs que les démos en elles-mêmes et même tellement bien, que New Mode pourrait les commercialiser sans problème. Une bonne raison de regarder la Syntax Terror Demo...

ST M. Certains ont baptisé cette démo, avant même sa sortie, la Union Demo 2. Qu'en pensez-vous ?

N. M. Non, je ne suis pas d'accord. La plupart des membres de The Union ont fait des écrans moyens. L'écran de TCB est même plutôt mauvais (si l'on oublie qu'il a été programmé en moins de 24 heures).



Chaos Inc, New Mode et Slime (Delta Force) assis. Oxygène (Lost Boys) préfère rester debout.

ST M. Des commentaires sur ses graphismes ?

Slime : Nous avons utilisé beaucoup de graphes d'autres groupes que nous avons ripés. Je n'en ai, en fait, dessiné qu'assez peu.

ST M. Pourquoi cette préférence pour Neochrome, et maintenant Neochrome Master, en graphisme ?

C. I. Ce programme a beaucoup de possibilités dont à la fois les graphistes et les programmeurs ont besoin. On ne peut pas, avec Neochrome, projeter des images sur une boule par exemple : c'est là un effet qui est agréable à regarder, mais que l'on utilise rarement en pratique. Par contre Neo Master a des possibilités qu'il est le seul à avoir, et qui en font le meilleur programme de dessin sur ST !

ST M. Quand as-tu commencé la reprogrammation de Neochrome ?

C. I. J'ai commencé à programmer Neochrome Master en mars 1990, et je travaille toujours dessus.

ST M. Mais Neo étant un programme réalisé par Atari Corp., ton entreprise est-elle légale ?

C. I. C'est un gros problème. J'ai écrit deux fois à Dave Staugas (l'auteur de la version originale) : dans mes lettres, je lui ai demandé l'autorisation de diffuser ma version améliorée. Je n'ai jamais eu de réponse. De même, j'ai donné à un grand responsable d'Atari une copie de Neo Master lors du salon de Düsseldorf cette année. Même chose : pas de nouvelles. Pas la moindre réprimande, pas le moindre signal de feu vert. Dans ma dernière lettre, j'ai demandé à Dave Staugas une réponse avant le 1er novembre (en lui laissant naturellement tout le temps nécessaire pour réagir), faute de quoi, je diffuserais mon programme. Dave Staugas a donc eu tout le loisir de faire régner ses droits ; il ne l'a pas fait (NDLR : nous profitons de cette occasion, et à l'invitation de l'auteur de Neochrome Master, pour en appeler aux responsables d'Atari : à quoi faut-il s'en tenir quant à ce programme ?).

ST M. Tu n'as pas l'intention de gagner de l'argent avec ?

C. I. Bien sûr que non : je n'en suis pas l'auteur, je l'ai seulement amélioré. Cela dit, je tiens à dire que la longueur originale de ce programme était de 58 ko et que la dernière version de Neochrome Master en est à 108 ko. Je crois donc pouvoir affir-



mer qu'au stade actuel, j'ai pratiquement tout reprogrammé : je suis maintenant l'auteur de la plus grosse partie de ce logiciel.

ST M. Quelles améliorations comptes-tu apporter encore à l'avenir ?

C. I. J'ai beaucoup d'idées en tête, mais je les programmerai uniquement si je juge l'intérêt des gens suffisant. Entre autres, une version TT complète (gérant les nouvelles résolutions), et peut-être une version monochrome.

ST M. Quelles études poursuivez-vous ?

C. I. Je suis à l'université, où je fais des études d'informatique. New Mode aussi. Slime est toujours à l'armée (rire général, sauf de la part de Slime !).
S. J'en ai encore pour plusieurs mois...

ST M. Quels sont vos projets futurs ?

N. M. Peut-être de nouvelle démos, mais je ne pense pas... Je ne sais pas... En tout cas, la programmation de jeux, comme tu le sais déjà.

ST M. Voulez-vous un exemplaire du ST Mag, quand il sortira ?

Tous en coeur : Bien sûr !

* * *

Cette interview de TEX a été réalisée dans la journée du dimanche 23 décembre 1990, en anglais et en allemand. Nous remercions les personnes suivantes, pour avoir bien voulu nous accorder une partie de leur temps : 6719, Es, Mad Max, Daryl et Ilja de Level 16 (maintenant chez TEX). A noter l'absence de -ME-, qui n'a pas pu se rendre en Hollande pour cause d'examens.

ST M. Vous êtes le meilleur exemple d'un groupe qui a d'abord créé des démos, et qui s'est ensuite tourné vers la programmation de jeux. Est-ce que vous considérez le fait d'avoir d'abord fait des démos, comme un avantage au moment de programmer des jeux ?



TEX - de g. à d.: 6719, Mad Max, ES, Daryl et Ilja (Level 16). Au centre : Charly, la mascotte vénérée !

ES. A mon sens, uniquement en ce qui concerne la technique. En effet, la jouabilité d'un jeu par exemple n'est pas quelque chose dont on s'occupe en programmant des démos. Il s'agit là d'un domaine le plus souvent inconnu au programmeur de démos, et qu'il nous a fallu maîtriser au plus vite.

ST M. L'AK-Screen dans la Life's a Bitch Demo a agréablement surpris tout le monde : cela faisait plus d'un an que l'on n'avait rien vu de vous. Aujourd'hui, dans la Syntax Terror Demo et 3 mois après, de nouveau un écran par TEX. Est-ce que c'est le début d'un retour plus fréquent à la programmation de démos ?

DARYL : Non, nous avons fait ces deux écrans juste pour le fun. Cela dit, rien ne nous empêche de continuer à faire une démo de temps à autre.

ST M. Combien de temps as-tu mis pour programmer ton écran dans la Syntax Terror Demo ?

D. Environ 8 mois, mais naturellement pas à temps complet : uniquement lorsque j'en avais le temps et l'envie !

ST M. Mad Max, te considères-tu comme un musicien ou comme un programmeur ?

ES : Oui ! (rire général)
MAD MAX : Musicien ! Mes connaissances en assembleur me sont ensuite très utiles pour programmer des musiques.

ST M. Qui est ton musicien favori sur ST ?

M. M. Je ne sais pas.

ST M. Count Zero ?

M. M. : ...

ES à **M. M**. : Count Zero, tu sais, celui du deuxième Maggie...

M. M. à **ES** (en riant) : Je n'en sais rien, je n'ai entendu sa musique qu'une seule fois.

ST M. Whittaker ?

ES : Non, pas Whittaker !

M. M. Beuh, certainement pas !

ES à **M. M**. : Tu devrais avoir une idée, pourtant !

M. M. Va pour Count Zero, mais aussi Ben Dalglish.

ST M. Et sur Amiga ?

M. M. J'aime beaucoup Chris Hülsbeck, Romeo Knight du groupe Red Sector et le musicien de Kefrens.

ST M. Quelle impression cela te fait-il de voir la plupart des groupes sur ST utiliser tes musiques, alors qu'ils ne devraient peut-être pas ?

M. M. Ça ne me fait rien !

ST M. Et que penses-tu des groupes qui ont fait des petits éditeurs de musiques basés sur tes routines ?

M. M. Sur ST ?! (Rire général !). Je n'étais pas au courant !

ST M. Tes éditeurs/ tes routines de musique sont-ils finis, ou essaies-tu au contraire de les améliorer en permanence ?

M. M. Ils évoluent en permanence, et depuis toujours.

ST M. ES, es-tu resté un vrai fan de Neo ?

ES. Oui, absolument, mais de la nouvelle version (par Delta Force). C'est aussi l'outil dont je me sers pour nos jeux, et il est à mon sens même mieux que (BIIIIP) sur Amiga.

Interviews réalisés par Patrick Raynaud et Klaus Berg.



2.5 Mo pour STF

"1ère GÉNÉRATION"

Nous allons ce mois-ci dépasser les 1 Mo d'un STF de la première génération (Rams sous le clavier). Pour les autres, patience : nous verrons bientôt un montage similaire, appliqué à la seconde génération (Rams sous l'alimentation et emplacement blitter)...

Les STF sont conçus pour un maximum de 1 Mo en deux bancs de 16 rams 256 Kbits (32 Ko). Pour étendre à plus d'un méga, il est nécessaire d'utiliser des Rams 1 Mbits (128 Ko), mais attention : ces dernières viendront se mettre en remplacement d'un des deux bancs déjà présents (la MMU ne pouvant en gérer plus).
A partir de là, se pose un problème pour ceux qui disposent déja d'1 Mo : il devient nécessaire de libérer un banc. Les possesseurs de 520 gonflé (520+) devront retirer les Rams des supports installés ; ils pourront cependant laisser en place leurs supports, comme nous allons le voir plus loin.

Pour les possesseurs de 1040, l'affaire est plus délicate, étant donné que les Rams du deuxième banc sont elles aussi soudées ; le choix oscillera entre dessouder ces 16 rams, opération réservée uniquement aux habitués du fer et du piston, ou laisser les rams en place et les déconnecter, l'extension venant par dessus. Dans ce dernier cas, la déconnection se fait en coupant l'alimentation de chaque Ram (broche 8) avec une pince coupante à bec très fin qu'il faut mieux emprunter qu'acheter (propre à vider une bourse).

Au sujet de l'ouverture du STF, un bref rappel : retirer les 7 vis et les 3 du lecteur (plus longues) ; enlever alors le capot supérieur, déconnecter le drive puis l'alimentation et retirer les 2 vis de fixation des deux pieds inférieurs (écartez le plastique au niveau du commutateur de tension si l'alimentation refuse de sortir...) ; bien entendu, vous débrancherez le clavier. Reste à retirer les 3 vis à la base du blindage ; vous pouvez alors sortir du plastique l'ensemble blindé, puis détordre les languettes métalliques afin de retirer la partie supérieure du blindage, laissant apparaître la carte mère (la première fois, ça impressionne !) que vous retirez à son tour du blindage inférieur.

L'extension consiste en une adaptation des broches des rams 1 Mbits aux trous de la carte mère prévus pour accueillir les rams 256 Kbits. Pour cela, j'ai réalisé un petit module d'adaptation qui reçoit 8 rams 1 Mbits.

Deux modules sont donc nécessaires pour l'extension de 2 Mo (16 x 128 Ko), ce qui portera la machine à 2.5 Mo avec le premier banc de 512 Ko ; mais il suffit de procéder de même sur ce premier banc (2 autres modules 1 Mo) pour monter à 4 Mo, soit le maximum permis par la MMU. Le tracé du module vous est fourni : vous remarquerez qu'il est simple face, ce qui facilite, sinon la gravure, du moins l'achat (30 frs environ, si vous passez par une boutique).

La figure 1 vous détaille l'implantation des composants. Pour les liaisons entre le module et la carte mère, vous allez devoir souder du même côté que la barette mâle/mâle, ce qui n'est pas évident (il faut une panne fine pour laisser un peu d'espace (1/2 mm) entre le cuivre et le plastique du support mâle/mâle. Veillez à bien mettre les broches les plus épaisses du côté soudure, puisque les plus fines sont destinées à s'enficher dans les supports du banc : seules les broches les plus fines s'enfichent dans des supports tulipes.

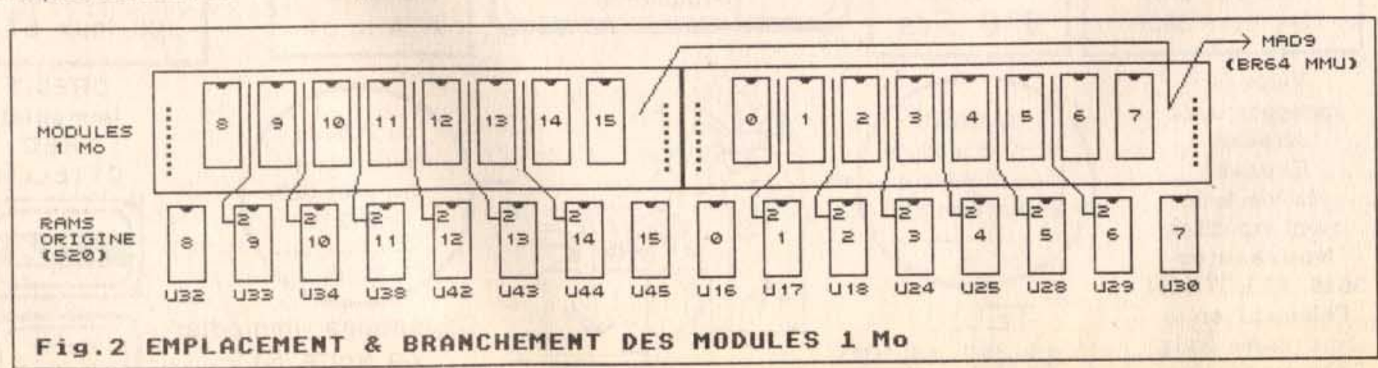


Fig.2 EMPLACEMENT & BRANCHEMENT DES MODULES 1 Mo

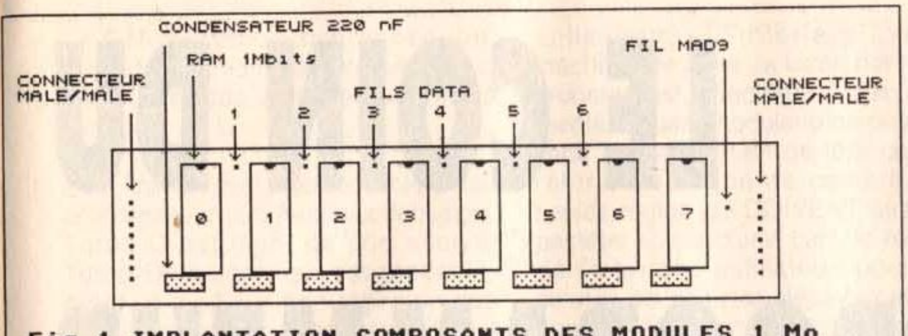


Fig.1 IMPLANTATION COMPOSANTS DES MODULES 1 Mo

Si vous avez dessoudé les rams (1040) ou les supports (520 étendu), ou si vous avez des supports lyres, le sens de ces supports mâle/mâle devient sans importance ; il en ira de même dans le cas d'un 520 non modifié, pour lequel vous aurez juste quelques trous à déboucher. D'ailleurs, les broches fines sont plus faciles à souder sur la petite platine (plus de place pour la panne !).

Pour les rams dynamiques, vous risquez, si vous avez gardé les supports sur la carte mère, de manquer de place en hauteur pour mettre des supports (tulipes !) sur la platine (à vérifier); auquel cas, vous devrez souder les rams sur la platine, ce qui ne pose pas de problème particulier, sauf si vous vous éternisez sur les soudures (surchauffe de la ram !). Restent les condensateurs de 220 nF à souder : choisir un modèle perle (de préférence) si vous risquez de manquer de place en hauteur. Souder ensuite les 6 fils et le fil MAD9, dont le bout viendra sur le MAD9 de la barrette de droite, un fil en repartant vers la MMU (voir figure 2). Si vous avez un 520 sans extension, il vous reste encore à déboucher deux rangées de trous pour chaque module. Maintenant, avant de continuer, il pourrait être sage de vérifier votre travail avec un Ohmètre, en particulier les supports mâle/mâle, dont les soudures ne sont pas toujours réussies (j'ai décelé sur mon montage une broche non soudée qui semblait pourtant l'être !).

A ce stade, il n'y a plus qu'à insérer les deux modules dans les trous et vérifier que rien ne touche entre la carte mère et les soudures des modules (logiquement, non!) puis, retourner la carte mère en tenant les modules (scotch), et souder.
Retournez de nouveau la carte mère : il vous reste à souder les 6 fils de chaque module sur la broche 2 des rams correspondantes (figure 2). Ces 6 fils relient les broches de donnée des rams 1 Mbits, les deux rams de chaque extrémité du module étant déjà connectées par les supports mâle/mâle. Une dernière soudure : le fil MAD9, partant du module de droite en direction de la broche MAD9 (64) de la MMU.

Pour cela, vous devez faire passer votre fil du côté soudure ; le mieux est d'utiliser le trou au-dessus du module de droite, destiné à laisser la fixation plastique. Attention à ne pas vous tromper de broche à souder lorsque vous retournez la carte mère (4ème broche en partant du coin : je vous laisse le soin de la trouver !). Il ne vous reste plus qu'à remonter la machine sommairement et vérifier qu'elle s'initialise correctement, ce qui devrait être le cas, si vous n'avez pas fait de bêtise... Si un quelconque logiciel de test ou autre venait à vous déclarer seulement 512 ko, il faudrait alors vérifier que tout est bien soudé (les fils).

Dans un prochain numéro, je vous proposerai donc la même extension pour les STF de la dernière génération (opération plus retorse) que vous êtes bien plus nombreux à posséder. Nous tenons enfin à remercier la société C.D.S.T.I. pour la CAO du module d'extension. Qu'on se le dise: toute utilisation du contenu de cet article, en particulier le tracé du montage, dans un but commercial est strictement interdite sans l'autorisation écrite de la société C.D.S.T.I. et de l'auteur.

Rodolphe Czuba

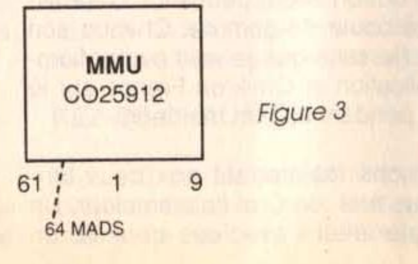


Figure 3



LE COIN DU PROGRAMMEUR

Voici enfin une rubrique que nous vous annoncions depuis longtemps. Elle est dédiée aux programmeurs de tout poil, débutants ou confirmés, quel que soit leur domaine ou leur langage de prédilection. Ici, nous discuterons des sujets les plus divers, avec, je le souhaite, votre participation. Mais pour commencer, il nous faut longuement disserter sur quelques mises au point de rigueur.

AVERTISSEMENT

ST Magazine tient à avertir ses lecteurs que l'abord des paragraphes suivants risque de nuire à la santé de personnes sensibles (i.e. qui programment en autre chose que du C ou de l'assembleur), et déconseillons à celles-ci leur lecture. Qu'on se le dise.

LANGAGES

Avant tout, parlons un peu des langages. Ils sont légion sur ST, mais trois catégories émergent de la masse :
- Les BASICs (GfA surtout, mais aussi Omikron) ;
- Les C, sur lesquels je vais revenir dans quelques instants ;
- Les Assembleurs.

Certains crient déjà au scandale en ne voyant pas le Pascal, le Modula-2 ou quelque autre langage dans notre petite liste : je considère en effet que ces langages n'émergent pas de la masse.

Au chapitre des langages "émergés" figure le GfA, au sommet de la vague. Je redoublerai ici de franchise et déclare sans ambages mon manque d'affection pour ce dernier. Les cris de réprobation redoublent. Rassurons quand même les populations : je m'incline devant le nombre en n'écartant pas systématiquement le GfA. Mais je serai clair : le GfA et autres BASICs, langages très appropriés pour "bricoler" un ou deux programmes ici ou là, ne me semblent en aucun cas des outils de développement sérieux. En dépit des dizaines de programmeurs qui se préparent à m'agiter à la face des listes entières de sources GfA d'excellente facture, je persiste à penser que les mêmes logiciels écrits en C auraient été largement meilleurs, plus rapides, plus petits, plus fiables.

Pour faire juste mesure, j'ajouterai un autre langage dans la colonne des mal-aimés, l'Omikron. Franchement, GfA ou Omikron, pour moi, c'est kif-kif et boule de gomme. Chacun son truc (je sens que je vais avoir Micro-Application et Omikron France sur le dos pendant un bon moment).

Passons maintenant aux deux langages rois : le C et l'assembleur. Un couple idéal : avec ces deux-là, on peut tout faire, usant en proportion variable de chacun suivant les besoins, du 100% C au pur 68000 .

En ce qui concerne le C, on relèvera deux catégories : les bons (K+R), parmi lesquels le Mark Williams C, le Laser C et le Lattice C 3, et les très bons (ANSI), catégorie qui ne regroupe, malheureusement, que 2 systèmes : le Turbo C et le Lattice C 5. Nous ignorerons bien sûr les mauvais et autres médiocres, en particulier l'antique Alcyon (kit de développement DRI/Atari).

Et maintenant, le coup de grâce... Je vous livre le nom de mon compilateur préféré : Turbo C. Encore des cris (je ne sais pas à quelle heure vous lirez cet article, mais gare au tapage nocturne); ce système, de plus, n'est pas disponible en France, et ne risque pas de l'être.

Je pense néanmoins que la qualité de ce compilateur vaut bien quelques efforts, l'achat en Allemagne n'étant quand même pas une épreuve insurmontable. Il sera en revanche plus difficile pour certains d'affronter la documentation en Allemand : vous pourrez alors vous reporter sur le Lattice C 5 de HiSoft, distribué en France par Human Technologies. Ces deux compilateurs répondant à la norme ANSI, la compatibilité devrait être assez bonne, et les listings donnés ici devraient passer directement.

Ensuite, les assembleurs. Le plus simple et pratique à la fois est pour le moment Devpac 2, distribué lui aussi par Human Technologies.

Le GfA-Assembleur propose, lui, quantité d'options supplémentaires, mais n'est pas d'une ergonomie à mon goût.

Pour finir, un outil indispensable à tout programmeur : un moniteur/debugger. Turbo C est muni de son acolyte Turbo Debugger, qui est probablement le meilleur debugger source que je connaisse, et qui est vraiment très pratique. Mais pour les opérations plus "en profondeur" (explorer la ROM, surveiller une interruption, etc.), il faut un outil un peu plus proche de la machine. Le premier et le plus classique est MonST2, fourni avec Devpac et le Lattice C 5, mais l'idéal est le génial A-Débog, qui est vraiment une tête au dessus.

Voilà, le tour des logiciels est bouclé. Après cet épisode plein de cris, d'indignation et de plaintes, passons aux autres outils indispensables à une bonne programmation : les livres de référence.

ZE BOUKSSES

Le premier livre vraiment INDISPENSABLE est le Livre du Développeur édité par Micro-Application. Il a encore beaucoup de lacunes, est loin d'être parfait et n'est en aucun cas un livre d'initiation, mais c'est le livre de référence sur le ST le plus complet et le mieux fait que je connaisse. Vraiment indispensable.

Ensuite, concernant le langage utilisé, tout dépend du langage en question, évidemment. Pour le C, un bon Kernighan et Ritchie (deuxième édition) est indispensable. Pour l'assembleur, il existe de nombreux ouvrages de référence concernant le 68000, mais on me souffle que "Mise en oeuvre du 68000" chez Sybex est le meilleur du lot.

Avec tout ça, vous devriez être parés. Mais il faut encore que je vous parle de quelques petites choses dont vous ne trouverez trace nulle part. Les fameuses "règles de programmation". Précisons tout d'abord leur nature...

LES REGLES

Je ne vous parlerai jamais que de deux types d'interactions : l'interface utilisateur, et l'interface avec la machine. Le reste, la façon dont vous programmez, structuré ou pas, commenté ou pas, modulaire ou pas, efficace ou pas... Rien de tout cela ne rentrera en ligne de compte. Les seules règles qui DOIVENT être respectées sont : d'une part, le respect de l'interface utilisateur, pour que celui-ci ne soit pas dépaysé et qu'il puisse utiliser le logiciel efficacement, et d'autre part, le respect de l'interface avec la machine, pour que vos programmes tournent sur n'importe quel ST, STE, TT voire compatible ST (on ne sait jamais !), quelle que soit la RAM disponible, le TOS installé, les disques présents (disquettes, disques durs, RAM-disques...), etc.

Comme je n'ai jamais trouvé la moindre trace de règles officielles émanant d'Atari, j'ai dû, au fur et à mesure de mes expériences, heureuses ou malheureuses, me fixer un certain nombre de règles. Il y aura toujours des (censuré) de détails mal documentés, ou auxquels ont n'avait pas pensé (cong !), qui feront que, dans un coin ou un autre, on dénichera bien quelque chose qui ne marchera pas sur un " TTE " ou un Mega-XY muni d'un TOS 5.7 mais, a priori, on a plus de chances que ça marche en suivant ces règles qu'en ne les suivant pas !

MAIS ...

Le problème du ST, c'est qu'il dispose d'énormément de possibilités, qui ne sont pas toujours accessibles de façon efficace par l'intermédiaire du système d'exploitation, quand elles sont accessibles ! Et il faut savoir à quel niveau prendre les règles de programmation. On n'imposera d'évidence pas les mêmes contraintes à un traitement de texte sous GEM et à une démo en full-screen avec un hardscroll !

Je vais donc essayer de distinguer plusieurs "niveaux" de compatibilité. A vous de choisir l'un de ceux-ci en fonction de vos besoins et de vos ambitions.

REZ-DE-CHAUSSÉE

Niveau 0, pas de compatibilité du tout. Là, vous faites ce que vous voulez, mais je ne veux pas en entendre parler. Ça doit rester chez vous, enfoui au fond d'un disque dur ou d'une disquette, et ne doit pas en sortir.
Je classe dans cette catégorie tous les programmes qui font appel à des adresses et fonctions non documentées ou non reconnues. Si vous trouvez amusant de poker et peeker ici ou là, parce qu'en prenant un moniteur vous aurez remarqué telle ou telle propriété de cette adresse, c'est votre problème, pas celui des autres. Tiens, au passage, si vous avez le livre "Trucs et astuces en GfA", jetez-le immédiatement, c'est rempli d'âneries de cet acabit.

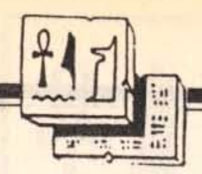
ENTRESOL

Ici, une petite catégorie spéciale, qui concerne les programmes basés sur des astuces non documentées, mais "reconnues". Il s'agit en particulier de tous les programmes faisant, avec plus ou moins de succès, de l'overscan, total ou partiel, et autres scrollings hardware sur STF (si, si, c'est possible). Ils sont basés sur quantité de bidouilles, de bugs de tel ou tel composant. Ça ne marche en aucun cas sur un TT ou toute machine future ou à venir, ça dépend de synchronisations qui ne pourront plus se faire si le processeur ou sa fréquence changent, mais c'est tellement génial, et représente un tel défi qu'on ne va tout de même pas empêcher les braves gens de faire des choses pareilles.

Par contre, par pitié, que ça reste dans le cadre de démos, ne vous avisez pas d'essayer de faire un programme commercial basé sur ces principes, c'est tellement incertain...

PREMIER ÉTAGE

Nous voici rendus au premier niveau de compatibilité. Ici, une différence marquante : on n'utilise que des adresses ou fonctions DOCUMENTÉES. Attention, documentées signifie "documentées par Atari", pas par le premier imbécile venu. Le commun des mortels ayant rarement accès à la documentation officielle complète d'Atari, il faut prendre avec des pincettes le contenu des livres, certains s'amusant à vous donner des adresses trouvées au détour d'une routine d'interruption.



Si je vous ai conseillé le livre du développeur, c'est justement parce que ce livre respecte les règles du jeu et, aux rares endroits où sont donnés des détails non documentés, on trouve une petite mention indiquant ce fait. Vous pouvez donc avoir confiance.

Par ailleurs, dans le cadre de cette rubrique, à moins que je ne devienne fou furieux ou irresponsable, vous ne trouverez ici que des adresses et fonctions documentées. Par contre, je ne peux malheureusement pas garantir que la totalité du magazine soit exempt d'astuces illégales, ses pages étant le fruit de la collaboration de plusieurs dizaines de personnes dont il est difficile de toujours vérifier en détail tout le travail ; j'espère que nos collaborateurs s'efforceront de suivre ces règles. Je vous donne les différentes catégories documentées dans l'encadré ci-dessous. En dehors de ces zones, point de salut !

- Registres d'E/S ($FF8000 à $FFFFFF) ;
- Zone des vecteurs d'exception ($0 à $3FF) ;
- Zone des variables systèmes documentées ($400 à $5A3 aux dernières nouvelles... Chaque nouvelle version du TOS agrandit un peu cette zone) ;
- Port cartouche ($FA0000 à $FBFFFF) ;
- Points d'entrée standard du TOS (Ligne-A, XBIOS, BIOS, GEMDOS, VDI, AES).

DEUXIEME ÉTAGE

Nous arrivons au niveau 2. Il s'agit de ne plus adresser directement les registres d'entrée-sortie, mais de passer par le XBIOS prévu à cet effet. Voici quelques-uns des cas dans lesquels ce niveau ne peut être atteint :
- gestion de palettes de couleur en interruption ;
- utilisation du son DMA ou du hardscroll sur STE (Atari n'a rien prévu de ce coté-là dans le TOS) ;
- remplacement de routines de gestion d'exception du TOS par des routines plus efficaces (en passant par les vecteurs documentés, bien sûr)...

Encore un détail à propos du hardscroll : le TT est censé reprendre toutes les caractéristiques du ST ET du STE ? Que nenni, point de hardscroll sur TT ! Et comment on fait pour savoir si on a le droit d'en faire ou pas ? On fait comme le dit Atari, on regarde l'adresse de la ROM ($FC0000 = ST, $E00000 = STE) ? Eh bien, pas de bol, la ROM du TT est en $E00000, merci Atari. Fin de la parenthèse.

A l'exception de ces cas et de quelques rares autres, on peut pratiquement tout faire en n'utilisant que le système d'exploitation, et les variables systèmes documentées ($400 et suivantes).

TROISIEME ÉTAGE

Il faut bien se rendre compte que certaines couches "basses" du système d'exploitation, en particulier la ligne-A et le XBIOS, sont très dépendantes de la machine : elles seront d'un usage assez difficile sur des machines ayant subi des extensions (cartes graphiques en particulier). Si vous voulez que votre programme marche sur TOUS les ST, STE, TT et autres machines portant le label "compatible ST" à venir (j'ai très peur du ST "de cartable" annoncé par Atari), ainsi que sur les machines ayant une extension de n'importe quel type, soyez sympas, évitez au maximum ces deux parties du TOS. Remarque en passant : la ligne-A n'est plus supportée par Atari à partir du TT. Elle y est encore, mais n'est pas garantie, en particulier dans le mode 256 couleurs, qui ne peut pas être géré par la ligne-A.

Evidemment, il existe encore de trop nombreux cas où on ne peut pas se passer du XBIOS. Au hasard :
- Le formatage d'une disquette. Suggestion à Atari : pourquoi ne pas prévoir dans l'AES une fonction "Formatage de disquette" ? Ça éviterait à de nombreux programmeurs de refaire mille et mille fois les mêmes routines, et permettrait de faire évoluer tous les programmes simultanément vers la haute densité par exemple...
- La configuration de l'interface série. Tiens, je vous mettrai le mois prochain un extrait de la doc développeur du TT, ça devient du pur délire. Là encore, si on gérait ça au niveau de l'AES, ce serait un gros progrès. En attendant, le plus simple est de laisser l'utilisateur passer par le panneau de contrôle.

D'un autre côté, les fonctions à éviter, ou au moins à surveiller de très près, du XBIOS :

- **Physbase/Setscreen** : dans certains cas, en particulier lors de l'utilisation d'une carte Matrix (et de nombreuses autres cartes grand écran ayant leur propre RAM vidéo), il est impossible de changer l'adresse physique de l'écran. On ne va pas vous empêcher de faire du switching d'écran, mais vérifiez bien que vous pouvez effectivement le faire (en faisant un Physbase après un Setscreen pour comparer, par exemple).

- **Getrez** : le plus grand piège de l'humanité. A proscrire. Evitez-la comme la peste. Pourquoi ? Parce qu'il est impossible de prévoir la réaction de cette fonction sur un écran "non-standard". Les programmes l'utilisant sont déjà surpris sur TT quand ils reçoivent une résolution "4" alors qu'ils sont habitués à avoir le choix entre 0, 1 et, 2, mais le plus marrant, c'est bien lors d'une rencontre avec une carte grand écran. Les concepteurs de ces cartes ne sachant pas quoi répondre à un Getrez, lui font généralement répondre 2. Quand on a du monochrome avec une résolution supérieure à 640x400, la plupart des programmes s'en tirent encore... en n'utilisant qu'une partie de l'écran. Mais quand en plus on a de la couleur... Je vous parlerai un peu plus tard de toutes les alternatives pour avoir les caractéristiques de l'écran.

D'autres fonctions du XBIOS sont plus incertaines quant à leur "sécurité", en général parce qu'on ne s'est pas encore trouvé confronté à un cas de figure qui rendrait leur usage impossible. Mais réfléchissez bien avant de les utiliser.
Un autre point d'incertitude concerne les variables de la ligne-A "à offsets négatifs". Il s'agit d'une collection de variables système documentées, obtenues en ajoutant un décalage

négatif à la base de variables de la ligne-A, qui rendent de fiers services. On y trouve en particulier une série de variables toutes bêtes, qui indiquent... la position de la souris ! La ligne-A n'étant officiellement plus supportée, on se demande ce qu'il en est pour ces variables, pourtant bigrement utiles. A priori, je pense qu'on peut considérer que la plupart d'entre elles sont encore utilisables. L'avenir nous le dira...

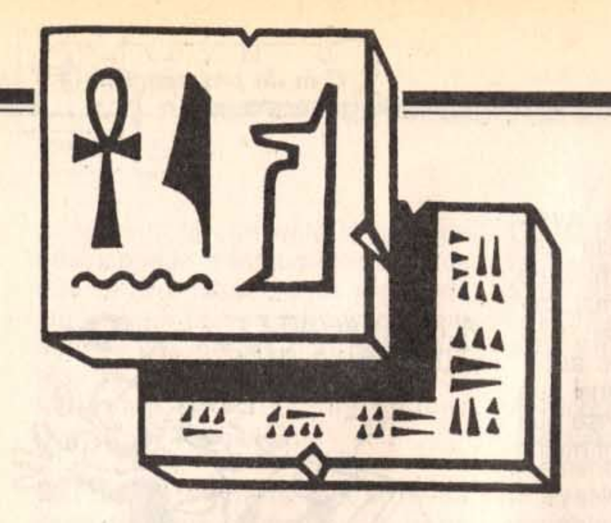
THE END

Voilà. Normalement, arrivés à ce niveau, un programme est censé avoir un comportement tout ce qu'il y a de plus normal sur tout ST, STE, TT... Nous parlerons le mois prochain du VDI et de l'AES, avec quelques conseils quant à leur utilisation, afin que le comportement ne soit plus "normal", mais optimal (utilisation de la totalité de l'écran, partage du temps avec d'autres applications ou accessoires...), ainsi que du TT. Rassurez-vous, je pense que les prochains épisodes seront moins "barbants" que celui-ci.

Si vous avez des questions, remarques, suggestions, et autres critiques à me faire, n'hésitez pas à me contacter, soit par courrier à la rédaction, soit par l'intermédiaire de notre serveur 3615 STMAG, en Bal STJC.

Jacques Caron





ANIMATION 3D (III)

Préparez-vous, car nous ne sommes pas au bout de nos peines. En effet, nous allons traiter ce mois-ci de la principale difficulté que nous puissions rencontrer : l'affichage complet d'un objet.

ENCORE DES MATHS...

J'entends déjà force grincements de dents dans le fond (près du radiateur), mais il est parfois des chemins parsemés d'embûches que l'on a plaisir à parcourir... Rassurez-vous, nous tâcherons d'être le plus clair possible.

Tout d'abord, la notation choisie pour la suite, si elle n'est pas très mathématique, évitera tous les problèmes éventuels d'impression. Ainsi, les vecteurs seront notés en majuscules (exemple : V1), sans la flèche qui devrait normalement les coiffer, mais que vous trouverez sur les figures. D'autre part, afin de différencier un vecteur de sa norme, nous noterons celle-ci entre parenthèses (exemple : (VN)). Cela dit, nous pouvons commencer.
Si vous vous remémorez le premier épisode de cette série, vous devez vous souvenir des trois axes que nous avions définis pour repérer un point dans l'espace (sinon recopiez-le 20 fois !). Nous avions donné pour chacun d'eux leur direction et leur sens. Il est important de distinguer ces deux termes, car ils ont une signification bien différente. Ainsi, les axes X, Y et Z ont pour direction respective l'horizontale, la verticale et la profondeur, et pour sens respectif de gauche à droite, de haut en bas, et de l'arrière vers l'avant. Le point O étant le point commun de ces trois axes, on le nomme origine du repère (O, X, Y, Z), repère qui va nous permettre d'orienter l'espace.

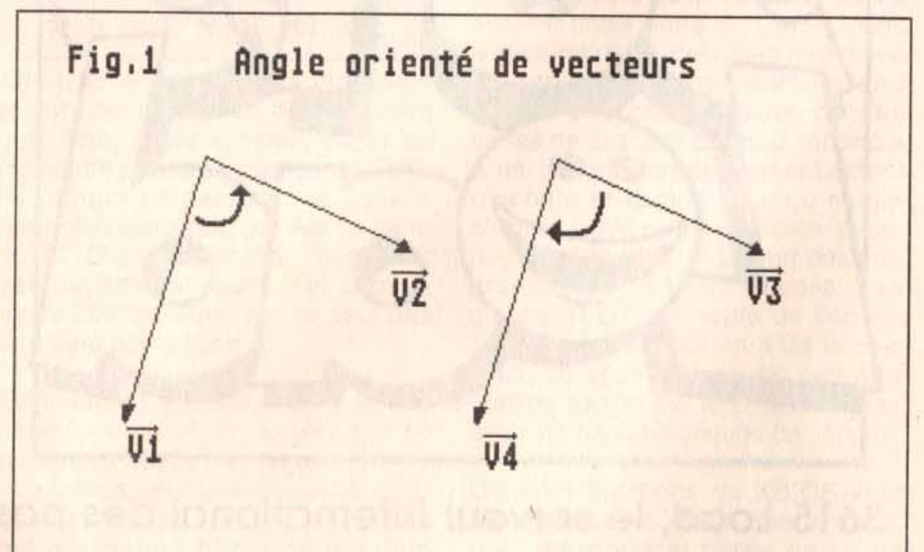
ORIENTONS L'ESPACE !

Tout vecteur de l'espace peut être défini par ses coordonnées dans notre repère, mais aussi par sa norme (sa longueur), et l'angle qu'il fait par rapport à chacun des axes ; orienter l'espace revient en fait à définir le signe de ces angles. En mathématique, un angle entre deux vecteurs V1 et V2 (qu'on notera "V1, V2"), est dit positif si l'on passe de V1 à V2 en tournant dans le sens trigonométrique (voir fig. 1). L'angle (V2, V1) est donc négatif.

Souvenez-vous simplement que le sens trigonométrique est en sens inverse des aiguilles d'une montre.

MAIS OU VEUT-IL EN VENIR ?

Les quelques remarques précédentes n'ont eu pour but que de nous amener dans le vif du sujet, à savoir la détermination des faces invisibles de notre objet. Un de nos soucis prioritaires étant la vitesse d'affichage, il va nous falloir constamment penser à ce qui va ralentir celle-ci, nous nous efforcerons donc en l'occurrence de n'afficher que les faces qui sont visibles par l'observateur. Si vous regardez un objet, vous n'en verrez en effet que les faces visibles (cela va sans dire...), les invisibles étant soit



L'angle (V1,V2) est positif: on tourne de V1 vers V2 dans le sens trigonomètrique. L'angle (V3,V4) est lui négatif: on tourne de V3 vers V4 dans les sens inverse du sens trigono--mètrique.

cachées par des faces plus proches, soit tournées vers l'arrière, c'est-à-dire le point de fuite (... mais ça va mieux en le disant). Attardons-nous au second cas, avant de voir que le premier sera résolu de façon bien plus simple.

DE L'INVISIBILITÉ D'UNE FACE

Le dernier paragraphe nous a innocemment donné la solution de notre problème. En effet, puisqu'une face qui n'est pas tournée vers l'observateur est invisible, il nous faut trouver le moyen de déterminer qu'une face est ou n'est pas tournée vers l'observateur, pour en déduire, si elle est ou n'est pas invisible. Et c'est là que notre petite leçon de maths va trouver usage. Prenons un objet simple, un cube par exemple (mais pas par hasard), et pour chacune de ses faces, prenons le vecteur qui lui est perpendiculaire et dirigé vers l'extérieur du cube. Ce vecteur, dit vecteur normal à la face et noté Vn, va nous donner la direction vers laquelle est tournée la face.

Ainsi, si le cube est le moniteur de votre ST, le vecteur normal à l'écran est dirigé vers vous, alors que le vecteur normal à la face arrière est dirigé vers le mur (ou ailleurs, qui sait). Décalez-vous maintenant un peu sur la gauche de votre moniteur : l'écran vous est toujours visible, bien que son vecteur normal ne soit plus directement dirigé vers vous, mais le côté gauche du moniteur vous est cette fois visible, même si là aussi son vecteur normal n'est pas directement dirigé vers vous (voir fig. 2). La face arrière persiste dans son invisibilité.

Fig.2

Vue de dessus

Oeil de l'observateur

Vue depuis l'observateur

Cas 1 / Cas 2

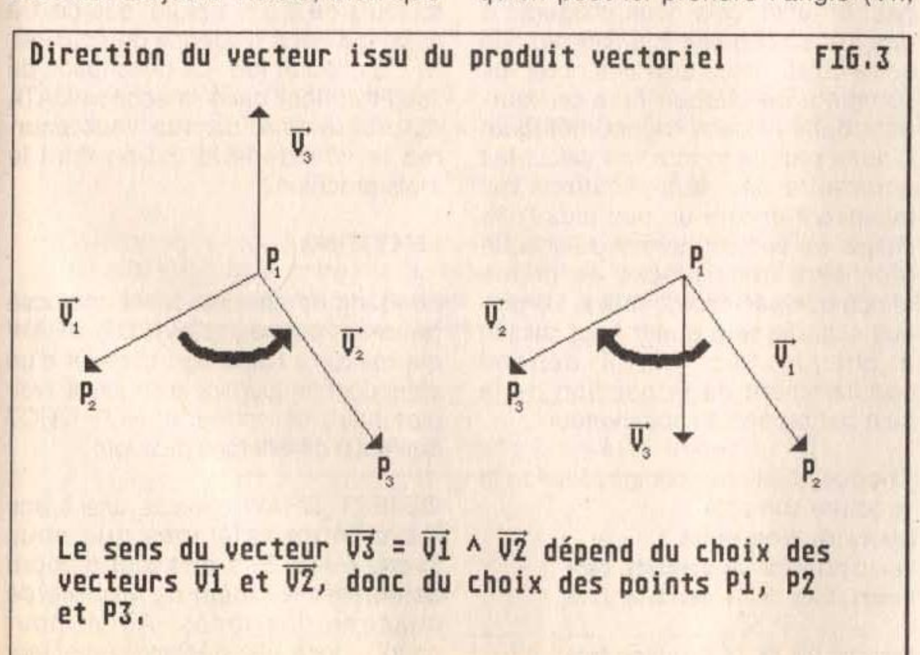
Cas 1:
θ1=0°
θ2=90°
θ4=180°
θ4=90°
Seule la face 1 est visible.

Cas 2:
θ1<90°
θ2>90°
θ3>90°
θ4<90°
Les faces 1 et 4 sont visibles.

Continuons ce raisonnement, et traçons à présent la droite qui relie chaque face à l'observateur, associons à cette droite un vecteur Vf qui a pour direction cette droite, et pour sens de la face vers l'observateur. Vous ne serez pas sans remarquer que pour qu'une face soit visible, il faut que l'angle (Vf, Vn) soit inférieur à 90 degrés (voir fig. 2). Remarquez qu'on peut ici prendre l'angle (Vn, Vf) qui a un signe opposé, car seule la valeur absolue de l'angle nous intéresse. Nous savons donc déterminer le critère d'invisibilité d'une face : il nous suffit de trouver l'angle que fait le vecteur normal à chaque face, avec le vecteur dirigé de celle-ci vers l'observateur. Et là encore, nos remarques préliminaires prennent tout leur sens (sans jeu de mot), bien qu'il nous fasse faire...

ENCORE UN PEU DE MATHS !

Une dernière fois, c'est promis. Donc, nous devons tout d'abord déterminer le vecteur normal de chaque face. Pour cela il nous faut deux vecteurs, donc 3 sommets P1, P2 et P3 de la face. On notera V1 le vecteur P1P2 et V2 le vecteur P1P3. Nous allons utiliser le produit vectoriel V1 ^ V2 qui aura pour résultat le vecteur V3, perpendiculaire à V1 et V2 (voir encadré). Le repère issu de V1, V2 et V3 est direct, les angles (V1, V2), (V2, V3) et (V3, V1) sont donc positifs. Si nous voulons que le vecteur V3 soit dirigé dans le bons sens (voir fig. 3), il nous faudra bien faire attention aux points que nous allons choisir.
Pour nous aider, il existe une règle très simple, appelée "Règle du Tire-Bouchon de Maxwell" par les électriciens (et "règle du bouchon de Malibu" par Pierre-Morel Fourrier) : le sens du vecteur V3 est le même que le sens de déplacement d'un tire-bouchon placé au-dessus du point P1 perpendiculairement à la face, et que l'on ferait tourner de V1 vers V2



Le sens du vecteur V3 = V1 ^ V2 dépend du choix des vecteurs V1 et V2, donc du choix des points P1, P2 et P3.

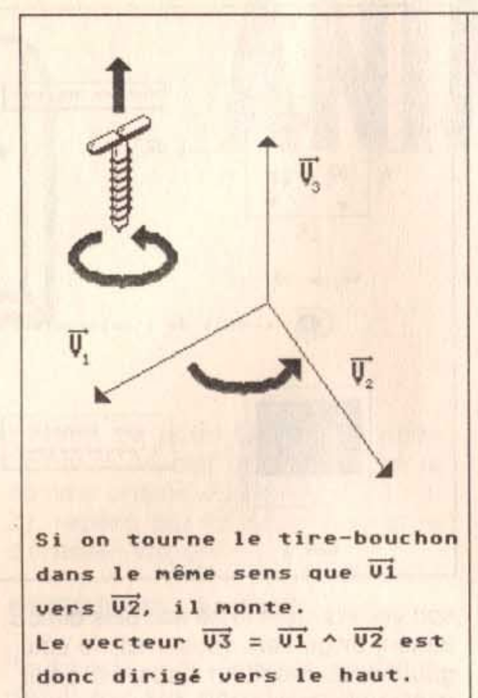


(voir fig. 4). Nous choisirons les points P1, P2 et P3, de telle sorte que le vecteur normal V3 "sorte" vers le côté visible de la face. Toutefois, cette règle ne s'applique que dans les systèmes droits, tels que les angles (I, J), (K, I) et (J, K) soient tous trois positifs. Or, notre système est gauche, puisque nous avons pris l'axe Y dirigé vers le bas, alors que seul l'angle (K, I) est positif. Dans le cas d'une face horizontale, nous devrons donc appliquer cette règle à l'envers et prendre le sens inverse du déplacement du tire-bouchon (rassurez-vous, nous verrons cela en détail le mois prochain).

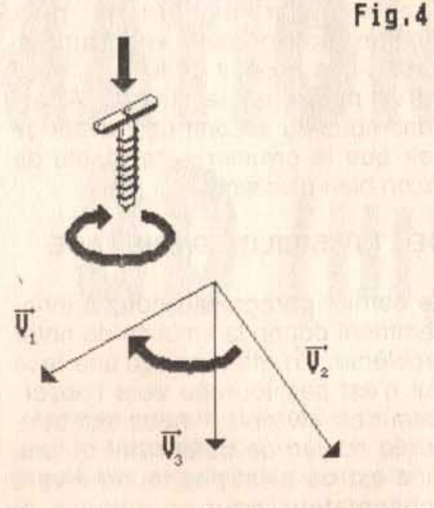
Nous avons maintenant le vecteur normal à notre face, il nous faut encore voir si celui-ci est dirigé vers l'observateur ou non. Nous allons ici utiliser le produit scalaire du vecteur normal Vn avec le vecteur dirigé d'un point de notre face vers l'observateur, qu'on notera Vf. L'encadré vous donne la formule permettant de calculer ce produit scalaire, et on peut remarquer que son signe dépend de l'angle entre les deux vecteurs (puisque leurs normes sont par définition positives). La remarque faite plus haut, selon laquelle le signe de l'angle nous est indifférent, vient du fait que le cosinus d'un angle et de son opposé sont égaux (cos(x) = cos(-x)). Comme nous l'avons dit plus haut, une face est visible si l'angle (Vf, Vn) est compris entre - 90 et + 90 degrés ; pour cette plage de valeurs, le cosinus est positif ou nul. On en déduit donc qu'une face est visible si le produit scalaire Vf.Vn est positif ou nul.

REVENONS AU PREMIER CAS

Nous savons désormais déterminer les faces invisibles de notre objet, nous ne les afficherons donc pas, mais il nous reste à voir les cas des faces partiellement visibles, cachées par d'autres faces. Comme nous l'avons dit auparavant, ceci va être résolu de façon très simple. Une face est partiellement visible si une autre face se trouve entre elle et l'observateur ; nous allons donc utiliser la méthode dite "du peintre", qui, lorsqu'il peint un paysage, commence par dessiner l'arrière-plan, puis dessine par-dessus les détails du premier plan. Nous allons trier les faces de façon à les afficher dans l'ordre de leur profondeur, en commençant par la plus lointaine, et en finissant par la plus proche de l'observateur.



Si on tourne le tire-bouchon dans le même sens que $\vec{V1}$ vers $\vec{V2}$, il monte. Le vecteur $\vec{V3} = \vec{V1} \wedge \vec{V2}$ est donc dirigé vers le haut.



Si on tourne le tire-bouchon dans le même sens que de $\vec{V2}$ vers $\vec{V1}$, il descend. Le vecteur $\vec{V3} = \vec{V2} \wedge \vec{V1}$ est donc dirigé vers le bas.

RÉALISATION PRATIQUE

Votre oeil perspicace a sans doute remarqué que nous avons été très peu bavard sur le choix des vecteurs que nous prendrons, pour calculer le vecteur normal de chaque face, ceci pour la simple, mais bonne raison, que le programme ne les calculera pas. En effet, cela nous obligerait à être très strict quant à la définition de notre objet : nous devrions noter les sommets de chaque face en tournant dans le sens trigonométrique. D'autre part, la formule de calcul fait apparaître des multiplications qui ralentiront encore un peu plus l'affichage. Le vecteur normal de chaque face sera donc stocké en même temps que ses coordonnées. Le produit scalaire sera quant à lui calculé à chaque fois, car il dépend constamment de la position de la face par rapport à l'observateur.

Chaque objet sera composé selon la structure suivante :

```
Nombre de faces moins 1
Description de la première face
Description de la deuxième face
...
Description de la dernière face
```

La description de chaque face se fera comme suit :

```
Nombre de sommets moins 1
Couleur (0-15)
Coord. du vecteur normal  (xn, yn, zn)
Coord. du premier sommet  (x1, y1, z1)
Coord. du deuxième sommet (x2, y2, z2)
...
Coord. du dernier sommet (xd, yd, zd)
```

Nous avons choisi de stocker le nombre de faces et de sommets décrémenté de 1, car les boucles effectuées par le programme se font toujours de 0 à n-1 plutôt que de 1 à n, ce qui nous dispense de soustraire 1 à chaque fois. La description de l'objet se fera dans la section DATA du programme, comme vous pourrez le voir dans le listing final le mois prochain.

LE LISTING

Le listing de ce mois-ci est composé de deux routines : OBJECT_DRAW qui réalisera l'affichage complet d'un objet dont on fournira la structure (voir plus haut) en entrée, et ROT_VECT que nous détaillerons plus loin.

OBJECT_DRAW effectue une à une les différentes étapes que nous avons vues. Elle doit tout d'abord déterminer le critère de visibilité de chacune des faces. Ainsi, pour chaque face elle déplace le vecteur

normal selon les angles x_angle, y_angle et z_angle (ROT_VECT), puis déplace son premier point (SP_TO_PL) afin de déterminer le vecteur dirigé de la face vers l'observateur. Le produit scalaire des deux vecteurs obtenus est alors calculé, et selon le signe du résultat, la face est déclarée comme visible ou non.

Si la face est visible, ses points sont à leur tour déplacés selon les mêmes angles, et les déplacements x_dep, y_dep et z_dep, et les coordonnées planaires résultantes placées dans le tableau COORD qui contiendra la description planaire de toutes les faces visibles (identique à la description spatiale vue plus haut, à la profondeur z près). L'adresse de cette structure est stockée dans le tableau COORD_TAB en vue du tri que nous effectuerons ensuite. Pour ce même tri, il nous faut calculer la profondeur moyenne (zm) de la face qui est ensuite placée dans le tableau ZM_TAB, et stocker le numéro de la face traitée dans le tableau NFACE_TAB.

Une fois toutes les faces visibles déterminées, peut s'effectuer le tri des faces selon leur profondeur moyenne croissante. L'algorithme que nous avons (très arbitrairement) choisi est celui du tri à bulle (Bubble-sort), qui n'est pas le plus rapide, mais simple à réaliser et à comprendre.

Le tableau NFACE_TAB est en fait trié parallèlement au tableau des profondeurs moyennes ZM_TAB. Ainsi, même si les descriptions des faces dans le tableau COORD ne sont pas triées, on y accédera dans l'ordre par l'intermédiaire du numéro de la face dans NFACE_TAB, et par son adresse stockée dans COORD_TAB. Les faces enfin triées, l'affichage se fait par la routine POLYGONE, en lui fournissant l'adresse de la structure de la face à afficher.

Revenons rapidement sur la routine ROT_VECT, qui s'inspire largement de la routine SP_TO_PL, mais se borne à effectuer les rotations désirées pour les vecteurs normaux. Il est inutile de déplacer les vecteurs normaux selon x_dep, y_dep et z_dep : cela ne changerait en rien leurs directions et sens. D'autre part, la projection de ces vecteurs sur le plan de l'écran ne nous est d'aucune utilité. Nous avons donc choisi de faire deux routines distinctes, pour un gain de vitesse appréciable lors des calculs.

PRODUIT VECTORIEL

Le produit vectoriel s'effectue entre deux vecteurs V1 et V2, et a pour résultat un troisième vecteur V3 qui est défini comme suit. Si les coordonnées respectives de V1, V2 et V3 sont (x1, y1, z1), (x2, y2, z2) et (x3, y3, z3), on a :

```
V1 ^ V2 = V3 (X3, Y3, Z3)

x3 = (y1 * z2) - (z1 * y2)
y3 = (z1 * x2) - (x1 * z2)
z3 = (x1 * y2) - (y1 * x2)
```

Cette formule est vraie dans un système droit, or notre système est gauche, car nous avons pris l'axe Y dirigé vers le bas. Nos formules deviennent alors :

```
x3 = (y1 * z2) - (z1 * y2)
y3 = (x1 * z2) - (z1 * x2)
z3 = (x1 * y2) - (y1 * x2)
```

Le vecteur V3 est perpendiculaire à V1 et à V2, ce qui nous servira pour déterminer le vecteur normal à une face.

PRODUIT SCALAIRE

Le produit scalaire s'effectue entre deux vecteurs, V1 et V2, et a pour résultat un scalaire (valeur réelle) qui est défini comme suit :

```
V1.V2 = (V1) * (V2) * cos (V1, V2)
```

Où (V1) et (V2) sont les normes de V1 et V2, c'est-à-dire leurs longueurs respectives. Si V1 et V2 ont pour coordonnées (x1, y1, z1) et (x2, y2, z2), on a la formule suivante :

```
V1.V2 = (x1 * x2) + (y1 * y2) + (z1 * z2)
```

Comme vous le montre la première définition, le signe du produit scalaire dépend de l'angle entre les deux vecteurs. En effet, la norme d'un vecteur étant par définition positive, seul le signe du cosinus intervient, comme le montre le tableau suivant :

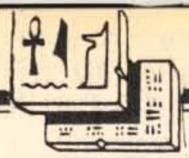
angle (V1, V2) en degrés	signe du cosinus
de 0 à 90	+
de 90 à 180	-
de 0 à -90	+
de -90 à -180	-

BONNE DIGESTION...

Ne vous en faites pas, si une partie de cet article vous a paru bien lourde à digérer, les formules sont à prendre pour ce qu'elles sont : des formules à utiliser, même si vous ne savez pas les démontrer. L'essentiel étant ici de bien comprendre la raison qui nous a poussé à utiliser certains outils mathématiques. Comme toujours, toute question ou remarque sera la bienvenue dans la BAL OYA SHIVO, sur le 3615 STMAG.

La suite sera, je vous le promets, bien moins théorique, car nous aborderons la denière étape que nous nous étions fixé, l'animation de notre objet, et nous verrons plus en détail comment définir celui-ci.

Benoit ARRIBART



```
;*************************************************
; rotation sans deplacement des vecteurs normaux *
; en entree: xe,ye,ze coordonnées du vecteur     *
;            x_angle,y_angle,z_angle             *
; en sortie: nouvelles coords dans new_vxe       *
;            new_vye et new_vze                  *
;*************************************************

rot_vect    movem.l d0-d6/a4,-(a7)
            lea xe,a4                   ;pointe coords
rot_v_loop
    move.w (a4)+,d0         ;xe
    move.w (a4)+,d1         ;ye
    move.w (a4)+,d2         ;ze
    move.w d0,d3            ;sauve coords
    move.w d1,d4
    move.w d2,d6
    lea var_3d,a4  ;pointe zone précalculs
    muls (a4),d3            ;c*xe
    muls 2(a4),d4           ;f*ye
    muls 4(a4),d6           ;i*ze
    add.l d4,d6
    add.l d3,d6             ;new_ze*512
    move.w d0,d3            ;xe
    move.w d1,d4            ;ye
    move.w d2,d5            ;ze

    muls 6(a4),d3           ;a*xe
    muls 8(a4),d4           ;d*ye
    muls 10(a4),d5          ;g*ze
    add.l d3,d5
    add.l d4,d5             ;new_xe*512
    muls 12(a4),d0          ;b*xe
    muls 14(a4),d1          ;e*ye
    muls 16(a4),d2          ;h*ze
    add.l d0,d2
    add.l d1,d2             ;new_ye*512
    moveq #9,d0             ;division par 512
    asr.l d0,d5             ;new_ze
    asr.l d0,d2             ;new_xe
    asr.l d0,d6             ;new_ye
    lea new_vxe,a4
    move.w d5,(a4)+         ;new_vxe
    move.w d2,(a4)+         ;new_vye
    move.w d6,(a4)+         ;new_vze

    movem.l (a7)+,d0-d6/a4
    rts

;**************************************
; procedure d'affichage d'objet      *
; en entree: a0=adresse de l'objet   *
;**************************************

object_draw movem.l d0-d7/a0/a2-a6,-(a7)
    bsr trigo               ;précalculs trigo
    move.w (a0)+,d7         ;d7=nbre faces -1
    lea coord_tab,a6        ;tableau des pointeurs sur
                            ;les structures coord
    lea coord,a5            ;tableau des structures coord
    lea nface_tab,a4 ;tables des Nos de faces (pour tri)
    lea zm_tab,a3               ;tableau des z moyens
    lea xp,a2                   ;coords planes
    moveq #0,d0                 ;numéro de la face

;*** Test de visibilité de la face ***

obj_draw0
    move.l (a0)+,xe        ;coords x/y du vecteur normal
    sne null_flag             ;test de nullité
    move.w (a0)+,ze           ;coord z du vecteur
    sne null_flag+1           ;test de nullité
    tst.w null_flag           ;si x,y et z nuls
    beq obj_null            ;pas de test de visibilité
    bsr rot_vect   ;effectue rotation du vecteur normal
obj_null
    move.l 4(a0),xe             ;coords du premier
    move.w 8(a0),ze             ;point dans xe,ye,ze
    bsr sp_to_pl                ;déplace le point
    tst.w null_flag
    beq.w obj_draw01
    move.w z_pt_fuite,d5
    sub.w new_ze,d5             ;(z_pt_fuite-new_ze)
    move.w new_vze,d6
    muls d5,d6                  ;*new_vze
    move.w new_vye,d5
    neg.w d5                    ;-new_vye
    muls new_ye,d5              ;*new_ye
    add.l d5,d6                 ;ajoute

    move.w new_vxe,d5
    neg.w d5                    ;-new_vxe
    muls new_xe,d5              ;*new_xe
    add.l d5,d6                 ;ajoute
    tst.l d6            ;teste signe du produit scalaire
    bge.s obj_draw01    ;positif ? oui continue
                        ;sinon face suivante
    move.w (a0),d6              ;nb sommets-1
    addq.w #1,d6                ;nb sommets
    move.w d6,d5
    add.w d6,d6                 ;nb sommets*2
    add.w d5,d6                 ;+nb sommets
    add.w d6,d6                 ;*2 ->nbsommets*6
    addq.w #4,a0                ;+4
    adda.w d6,a0                ;pointe face suivante
    dbra d7,obj_draw0           ;face suivante
    bra.s obj_suite             ;passe au tri

;*** création structure coord de la face ***

obj_draw01
    move.l a5,(a6)+ ;sauve adresse de la structure coord
    move.w (a0)+,d6             ;nb sommets -1
    move.w d6,(a5)+             ;dans coord
    move.w (a0)+,(a5)+          ;couleur dans coord
    move.w d6,d5
    addq.w #1,d5          ;nb sommets pour calcul z moyen
    move.w new_ze,d1      ;z moyen initialisé avec z
                          ;du premier point
    move.w d0,(a4)+       ;stocke numéro de face
    move.l xp,(a5)+       ;xp,yp du 1er point dans coord
    addq.w #4,a0          ;pointe 2ème point
    subq.w #1,d6          ;nb points -2
```

```
obj_draw1
    move.l (a0)+,xe          ;x et y dans xe,ye
    move.w (a0)+,ze          ;z dans ze
    bsr sp_to_pl             ;déplace le point
    move.l (a2),(a5)+        ;stocke xp,yp dans coord
    add.w 8(a2),d1           ;somme des new_ze
    dbra d6,obj_draw1        ;sommet suivant
    ext.l d1
    divs d5,d1               ;calcule z moyen
    move.w d1,(a3)+          ;stocke z moyen
    addq.w #1,d0             ;numéro face +1
    dbra d7,obj_draw0        ;face suivante

;*** tri des faces sur z moyen croissant ***
obj_suite
    lea nface_tab,a4        ;tableau des numéros de face
    lea zm_tab,a3           ;tableau des z moyens
    subq.w #1,d0    ;limite recherche =nbre de faces -1
    beq.s obj_one           ;une seule face
    bmi.s obj_dr_end        ;aucune face
    move.w d0,d7            ;stocke nb faces

obj_sort0
    moveq #0,d1              ;marque
    moveq #0,d2              ;i=0
    moveq #0,d3              ;offset i

obj_sort1
    move.w 0(a3,d3.w),d4            ;d4=zm(i)
    move.w 2(a3,d3.w),d5            ;d5=zm(i+1)
    cmp.w d4,d5                     ;zm(i)<zm(i+1) ?
    bge.s obj_sort2                 ;non
    move.w d5,0(a3,d3.w)            ;échange zm(i)
    move.w d4,2(a3,d3.w)            ;et zm(i+1)
    move.w 0(a4,d3.w),d4            ;échange nface(i)
    move.w 2(a4,d3.w),0(a4,d3.w)  ;et nface(i+1)
    move.w d4,2(a4,d3.w)
    move.w d2,d1                    ;marque=i

obj_sort2
    addq.w #2,d3             ;offset i
    addq.w #1,d2             ;i
    cmp.w d2,d0              ;limite recherche>i ?
    bgt.s obj_sort1          ;oui
    move.w d1,d0             ;limite recherche = marque
    bgt.s obj_sort0         ;tant que limite recherche>0

;*** affichage des faces dans l'ordre des z moyens ***
    lea coord_tab,a6     ;adr. structures coords triées
obj_draw2
    move.w (a4)+,d0          ;numéro de la face
    lsl.w #2,d0              ;nface*4
    move.l 0(a6,d0.w),a5     ;pointe la structure coord
    lea nb_sommets,a0  ;pointe variables pour affichage
    move.w (a5)+,d0          ,d0=nb_sommets
    move.w d0,(a0)+          ;stocke nb_sommets
    move.w (a5)+,couleur     ;et couleur

obj_draw3
    move.l (a5)+,(a0)+     ;stocke xp et yp dans sommets
    dbra d0,obj_draw3      ;prochain sommet
    bsr polygone           ;affiche polygone
    dbra d7,obj_draw2      ;prochaine face

obj_dr_end
    movem.l (a7)+,d0-d7/a0/a2-a6
    rts

;*** une seule face: pas de tri ***

obj_one
    lea coord,a5             ;pointe structures coords
    lea nb_sommets,a0  ;pointe variables pour affichage
    move.w (a5)+,d0          ,d0=nb_sommets
    move.w d0,(a0)+          ;stocke nb_sommets
    move.w (a5)+,couleur     ;et couleur

obj_one1
    move.l (a5)+,(a0)+     ;stocke xp et yp dans sommets
    dbra d0,obj_one1       ;prochain sommet
    bsr polygone           ;affiche polygone
    movem.l (a7)+,d0-d7/a0/a2-a6
    rts

;*****************************************************
; la section bss est comme d'habitude à considérer *
; comme purement documentaire, elle sera reprise    *
; dans sa totalité dans le listing final.          *
;*****************************************************
            section bss

interc      equ 0       ;/* private joke */
nb_sommets  ds.w 1      ;nb sommets du polygone -1
couleur     ds.w 1      ;couleur du polygone

xe          ds.w 1      ;coordonnées spatiales
ye          ds.w 1
ze          ds.w 1
xp          ds.w 1      ;coordonnées planes
yp          ds.w 1

new_xe      ds.w 1      ;nouvelles coords après
new_ye      ds.w 1      ;rotation et
new_ze      ds.w 1      ;deplacement
new_vxe      ds.w 1     ;nouvelles coords du
new_vye      ds.w 1     ;vecteur normal
new_vze      ds.w 1     ;après rotation

x_angle     ds.w 1      ;angles de rotation
y_angle     ds.w 1
z_angle     ds.w 1
null_flag   ds.w 1      ;indic.de vecteur normal nul
z_pt_fuite  ds.w 1      ;profondeur point fuite
var_3d      ds.w 9      ;variables calcul 3d
zm_tab      ds.w 64     ;tableau des profondeurs
                        ;moyennes pour tri
nface_tab   ds.w 64     ;numéros des faces triées
coord_tab   ds.l 64     ;adresses pointant sur des
                        ;structures coord comme suit
coord       ds.w 1024   ;mot 0: nombre de sommets -1
                        ;mot 1: couleur / mot 2: x1,y1
                ;mot 3: x2,y2 / mot i*2+2: xi,yi ...etc
            end
```

PROGRAMMER EN OMIKRON (XI)

4.3. LES FONCTIONS SYSTEME DE L'ATARI ST

4.3.1. LEUR NATURE, LEUR UTILITÉ

On appelle fonctions systèmes toutes les fonctions qui sont dans la ROM de l'Atari ST (l'ensemble de toutes ces fonctions composent le système d'exploitation). On y trouve une grande variété de fonctions, allant de la lecture d'un fichier sur disque, à l'initialisation de l'horloge, en passant par la fonction permettant d'obtenir l'adresse de l'écran !

A priori, on pourrait penser que ces fonctions sont inutiles car on peut très bien réaliser toutes ces opérations grâce aux instructions de l'OMIKRON. C'est vrai pour une grande partie d'entre elles, par exemple il n'est pas nécessaire de passer par le système d'exploitation pour initialiser l'heure, il suffit de taper:

```
TIME$="20:31:10"
```

Mais, il existe des fonctions systèmes qui n'ont pas d'équivalent en OMIKRON, souvent parce qu'elles sont peu utilisées ou par souci de gain de place. Certaines manipulation ne sont réalisables que grâce à ce que nous allons voir dans les chapitres suivants.

Les fonctions systèmes marchent comme des procédures, c'est à dire que l'on peut leur passer des paramètres et elle peuvent retourner un paramètre. Par exemple, le paramètre à passer à la fonction permettant d'initialiser l'heure sera l'heure elle même et cette fonction nous retournera 0 si tout s'est bien passé et -1 sinon.

Ces fonctions système sont divisées en trois groupes de fonctions:

- GEMDOS ;
- BIOS ;
- XBIOS.

Dans les trois chapitres qui viennent, nous allons etudier les fonctions systèmes, ainsi que leur syntaxe et des exemples.

Cependant nous ferons la distinction entre les fonctions qui vous seront le plus utile (celles qui n'ont pas d'équivalent exact en OMIKRON) et celles qui vous seront inutiles, ces dernières ne seront pas mentionnées.

4.3.2. LES FONCTIONS GEMDOS

Voici comment vont se présenter les fonctions qui vont être décrites ici:

<Numéro de la fonction>: <Nom de la fonction>
Description de ce qu'elle fait
Paramètres d'appel: <liste des paramètres>
Paramètre de retour: <paramètre>

Par exemple, le nom de la fonction pourra être Pterm0() si il s'agit de la fonction Pterm0 et qu'il n'y a pas de paramètres ou bien Dfree(a,b) si la fonction s'appelle Dfree et qu'il faut lui passer deux paramètres.

Chaque fonction, en plus d'avoir un nom, a un numéro. C'est ce qui va vous permettre de les appeler depuis le basic. Prenons comme exemple une fonction quelconque:

```
$2: Cconout(A)
Cette fonction affiche un caractère à l'écran, elle
correspond a PRINT CHR$(a).
Paramètres d'appel: Le code Ascii du caractère
Paramètre de retour: Aucun
```

Pour appeler cette fonction il faudra faire:

```
GEMDOS(Retour,2,A)
```

L'instruction GEMDOS sert à appeler une fonction Gemdos. Le premier paramètre est la variable dans laquelle va être stocké le paramètre de retour, le deuxième paramètre est le numéro de la fonction à appeler, les paramètres suivants sont les paramètres à passer à la fonction système.

On pourra faire:

```
GEMDOS(Ret,2,65)
```

Cet appel GEMDOS correspond à la fonction 2 (qui est Cconout) et lui passe le paramètre 65 (65 est le code Ascii de la lettre A). Le paramètre de retour sera stocké dans la variable Ret.

Cependant, il y a un petit problème. En effet nous avons vu que la fonction Cconout (celle que nous appelons) n'a pas de paramètre de retour. Dans ce cas, la variable indiquée pour stocker le paramètre de retour n'est pas modifiée. Puisqu'elle ne sert à rien, on peut donc supprimer cette variable.

Attention, il ne faudra ommettre la variable retour uniquement quand il n'y a pas de paramètre de retour ou si le paramètre de retour ne nous intéresse pas.

L'exemple précédent sera équivalent à:

```
GEMDOS(,2,65)
```

Attention: n'oubliez surtout pas dans ce cas la virgule juste après la parenthèse, c'est elle qui indique que nous avons volontairement omis la variable pour le paramètre de retour.

Nous pouvons maintenant commencer l'inventaire des fonctions intéressantes du Gemdos !

```
$3: Cauxin()
Lit un caractère sur la Rs232
Paramètres d'appel: Aucun
Paramètre de retour: caractère lu
```

Cette fonction lit un caractère sur la Rs232 (la prise modem). La plupart du temps, cette interface sera utilisée pour communiquer avec un minitel. Nous allons voir maintenant les autres fonctions qui concernent cette interface puis nous verrons un exemple qui englobera toutes ces fonctions.

```
$4: Cauxout(a)
Envoie un caractère sur la Rs232
Paramètres d'appel: Code Ascii du Caractère
Paramètre de retour: Aucun

$12: Cauxis()
Demande l'état du tampon d'entrée de la Rs232
Paramètres d'appel: Aucun
Paramètre de retour: état du tampon
```

Le moment est venu de vous expliquer plus en détail le fonctionnement de l'interface série Rs232. Vous pouvez aussi bien envoyer que recevoir des caractères (en fait, des nombres de 0 a 255). Tous les envois seront faits à une certaine vitesse. La vitesse se mesure en bauds c'est à dire en bits par seconde (une vitesse de 1200 bauds correspondra à une vitesse de 150 caractères par secondes car il y a 8 bits dans un octet) et elle dépend du périphérique avec lequel vous communiquez. Quand il s'agit d'un minitel, la vitesse sera de 1200 bauds. Ainsi si la vitesse est de 1200 bauds et que vous envoyez à la suite 300 caractères, l'Atari va effectuer cette tache pratiquement instantanément, alors que les 300 caractères prendront 2 secondes a être envoyés. L'Atari va donc organiser une véritable 'file d'attente' que l'on appelera buffer.

Le buffer contient tous les caractères prêts à être envoyés. Le même processus est employé pour les caractères que l'Atari recoit, ces caractères sont en effet stockés eux aussi dans un buffer. A chaque fois que vous prendrez un caractère (à l'aide de la fonction Cauxin), l'Atari vous retournera le premier caractère reçu puis effacera celui-ci du buffer. Voici un exemple:

Si le minitel vous envoie les caractères qui ont pour code Ascii 65 puis 87 puis 66 puis 13 puis 67, le buffer d'entrée de la Rs232 se présentera comme ceci:

```
65
87
66
13
67
```

Si ensuite vous utilisez la fonction Cauxin() l'Atari va vous retourner 65 et le buffer se présentera comme ceci:

```
87
66
13
67
```

Si le buffer est vide et que vous faites un Cauxin() alors l'Atari va attendre indéfiniment un caractère de la Rs avant de vous rendre la main. Voila pourquoi il est quasiment indispensable de regarder si il y a bien un ou plusieurs caractères en attente grâce a la fonction Cauxis(). Voila comment pourra se présenter la partie de programme qui prend un caractère de la Rs:

```
REPEAT
   GEMDOS(Etat,$12)
UNTIL Etat<>0
GEMDOS(Caractere,3)
```

Ce programme va attendre jusqu'à ce qu'un caractère soit présent dans le buffer. Ensuite il va prendre ce caractère et le stocker dans la variable Caractere.

Exercice : Faites un programme qui vide le buffer d'entrée de la Rs (C'est à dire qui va prendre les caractères en attente jusqu'à ce qu'il n'y en ait plus)

Réponse:

```
GEMDOS(Etat,$12)
WHILE Etat=-1
  GEMDOS(,3)
  GEMDOS(Etat,$12)
WEND
```

Ce programme regarde d'abord l'état du buffer d'entrée et stocke cet état dans la variable Etat. Ensuite, il boucle tant que l'état est à -1 (vrai), c'est à dire tant qu'il y a un caractère en attente. Dans cette boucle, il va prendre le caractère qui est en attente et regarder à nouveau l'état.

Remarque à propos du minitel : Pour connecter votre ST à votre minitel, il vous faut tout d'abord le cordon adéquat. Ce cordon est en vente à la boutique de Pressimage.

Mais cela ne suffit pas, il faut également configurer l'interface Rs232 (c'est à dire indiquer à l'Atari 'comment' il faut communiquer avec le minitel).

Vous pourrez trouver l'instruction qui permet de réaliser cette configuration dans le chapitre sur les fonctions XBIOS, la voici cependant en 'avant-première' pour vous permettre d'utiliser les fonctions Cauxin, Cauxout et Cauxis.

Vous devrez mettre cette instruction avant d'utiliser une de ces trois fonctions, il est cependant inutile de la répeter à chaque fois, une seule fois suffit, à l'initialisation.

```
XBIOS(,$F,7,0,%10101110,1,1,0)
```

Exercice:

Ecrivez un programme qui permettra d'utiliser le clavier de l'Atari à la place de celui du minitel.

Le programme ne s'occupera pas des touches spéciales du minitel sauf ENVOI qui sera émulé par la touche de l'Atari [RETURN].

Pour faire un ENVOI sur le minitel, il faut envoyer à ce dernier les caractères ayant pour codes Ascii $13 et $41. Et n'oubliez pas que le code Ascii de la touche [RETURN] est 13.

Réponse:

```
REPEAT
  REPEAT
    A$=INKEY$
  UNTIL A$<>""

  Touche$=Mid$(INKEY$,4,1)
  IF Touche$=CHR$(13) THEN
    GEMDOS(,4,$13)
    GEMDOS(,4,$41)
  ELSE
    GEMDOS(,4,ASC(Touche$))
  ENDIF
UNTIL 0=1
```

Ce programme est composé d'une boucle sans fin (REPEAT...UNTIL 0=1). Dans cette boucle, le programme va tout d'abord attendre qu'une touche de l'Atari soit pressée, puis il mettre dans la variable Touche$ le code Ascii du caractère qui a été tapé.

Si ce code est 13, alors la touche tapée est [RETURN] et il faut envoyer la séquence $13 $41 pour émuler la touche ENVOI du minitel. Si, par contre, le code est différent de 13, alors c'est une autre touche qui a été frappée donc on envoie tout simplement au minitel le code Ascii du caractère tapé.

Poursuivons donc maintenant notre inventaire des fonctions GEMDOS !

```
$19 Dgetdrv(a)
Recherche le lecteur de disque par défaut.
Paramètres d'appel: aucun.
Paramètres de retour: 0 si disque A, 1 si B etc...
```

Cette fonction permet de trouver le lecteur de disque par défaut, c'est à dire le lecteur de disque sur lequel cherche l'Atari quand vous ne lui indiquerez pas de drive en particulier. Comment l'utiliser:

Exemple:

```
GEMDOS(Drv,$19)
PRINT "Le drive est: ";CHR$(65+Drv)
```

Ce petit programme va afficher sous la forme d'une lettre le drive courant (pour ceux qui ne s'en rapellent plus, CHR$ est l'inverse de Ascii et ici va nous donner la lettre de l'alphabet qui a le numéro Drv (A étant le zero puisque son code Ascii est 65)).

Bien sûr, nous n'avons vu ici que quelques fonctions du Gemdos, mais les autres sont très spécifiques (et par conséquence vous n'aurez à les utiliser que très rarement si ce n'est jamais) ou comportent un équivalent direct en omikron, dans ce cas, pourquoi se compliquer la vie ? Servons nous de l'OMIKRON !

4.3.3. LES FONCTIONS BIOS

Les fonctions BIOS sont les moins nombreuses, cependant certaines peuvent s'averer très utiles. Voici la liste de celles dont vous serez susceptible de vous servir :

```
$1: Bconstat(a)
Demande l'état d'un périphérique en entrée
Paramètres d'appel: code du périphérique
Paramètres de retour: -1 si pret, 0 sinon
```

Cette fonction est l'équivalent de Cauxis mais pour tous les périphériques. Nous avions vu que Cauxis demandait l'état du tampon de l'interface série Rs232c, Bconstat fait la même chose pour un périphérique que l'on spécifie.

Voici la liste de ces périphériques:

0: Imprimante (Inutilisé car pas d'entrée !)
1: Entrée-sortie serie Rs232c
2: Clavier
3: Prises MIDI
4: processeur clavier
5: Ecran (inutilisé car pas d'entrée !)

ATTENTION: la manipulation du périphérique 4 nécessite une connaissance du processeur clavier 6301.

La ligne: `BIOS(Ret,1,1)`

Sera equivalente à la ligne: `GEMDOS(ret,$13)`

Car 1 représente le périphérique Série Rs232c et la fonction $13 du Gemdos est (comme nous l'avons vu) Cauxxos qui demande l'état du tampon Rs232c.

```
$2: Bconin(a)
Attend l'entrée d'un caractère.
Paramètres d'appel: Code du périphérique.
Paramètres de retour: caractère.
```

Cette fonction fait exactement la même chose que la fonction du Gemdos Bconin mais pour n'importe quel périphérique.

```
$3: Bconout(a,b)
Envoie un caractère sur un périphérique.
Paramètres d'appel: Code du périphérique
                    Caractère a envoyer.
Paramètres de retour: aucun.
```

Bconout envoie un caractère tout comme la fonction du Gemdos Cconout mais sur n'importe quel périphérique.

exemple: La fonction

```
BIOS(,$3,1,65)
```

est équivalente à :

```
GEMDOS(,2,65)
```

```
$9: Mediach(a)
Regarde si la disquette du lecteur a été changée.
Paramètres d'appel: 0 pour lecteur A, 1 pour B...
Paramètres de retour: 0 si la disquette n'a pas été changée,
1 peut être, 2 si elle a été changée.
```

Cette fonction permet de vérifier si la disquette qui se trouvait dans le lecteur specifié a été retirée et remplacée (par une autre ou la même!).

Ceci est très utile si vous lisez le directory d'une disquette dans un programme. Si la disquette a été changée ou peut-être, alors il faut relire le repertoire, sinon c'est inutile et vous économisez donc du temps à l'utilisateur du programme.

```
$A: Drvmap()
Donne la liste des disques actifs.
Paramètres d'appel: aucun.
Paramètres de retour: un mot (deux octets).
```

Dans les deux octets qui sont retournés, le bit 0 est à 1 si le lecteur A est actif, le bit 1 est à 1 si le lecteur B est actif, etc... jusqu'au bit 15 pour le lecteur P.

Remarque: Il existe un moyen simple de tester si un bit donné d'un mot est à un, il suffit d'utiliser l'instruction BIT (référez-vous au manuel de l'OMIKRON pour la syntaxe)

4.3.4. LES FONCTIONS XBIOS

Voici en fait la grosse partie de ce chapitre sur les fonctions système. En effet, c'est dans les fonctions XBIOS que l'on trouve les plus pratique et les plus utilisées. Voici notre inventaire !

```
$2: Physbase()
Donne l'adresse de la base physique de l'écran.
Paramètres d'appel: aucun.
Paramètres de retour: Adresse.
```

Nous avons vu que l'écran se trouvait à une certaine adresse dans la mémoire, voici l'instruction qui permet de récuperer cette adresse. On l'appelle adresse physique car cette fonction retourne l'adresse de la base de l'écran que l'on voit.

```
$3: Logbase()
Donne l'adresse de la base logique de l'écran.
Paramètres d'appel: aucun.
Paramètres de retour: adresse.
```

Ici il s'agit de la base logique, c'est à dire l'écran sur lequel vont s'effectuer les opérations graphiques.

```
$4: Getrez()
Retourne la résolution de l'écran.
Paramètres d'appel: aucun.
Paramètres de retour: 2 haute, 1 moyenne, 0 basse.
```

Permet de regarder dans quelle résolution se trouve l'écran graphique de l'atari.

Exemple::

```
XBIOS(resolution,4)
```

ceci va mettre dans la variable Resolution la résolution de l'écran (2 haute, 1 moyenne . . .)

```
$5: Setscreen(a,b,c)
Fixe les configurations de l'écran.
Paramètres d'appel: Logbase,
                    Physbase,
                    Resolution.
Paramètres de retour: aucun.
```

Attention: On se sert très rarement de cette fonction pour changer de résolution. En effet, sur le ST, le changement de résolution est une opération très délicate (RESET). On réservera donc cette fonction aux changements de bases logiques et physiques de l'écran.

```
$14: Rsconf(a,b,c,d,e,f)
Configure le port Rs232c.
Paramètres d'appel: Vitesse,
                    Protocole,
                    Registre de contrôle general,
                    Registre de contrôle recepteur,
                    Registre de contrôle émetteur,
                    Synchronisation.
Paramètres de retour: inutile.
```

Cette fonction étant assez complexe, nous ne détailleront pas ici les différents paramètres. Nous vons avons donné la configuration la plus courante (minitel) précédemment, voici les différents codes de vitesse:

```
0: 19200      8: 600
1: 9600       9: 300
2: 4800      $A: 200
3: 3840      $B: 150
4: 2400      $C: 134
5: 1920      $D: 110
6: 1745      $E: 75
7: 1200      $F: 50
```

Voila donc notre inventaire des fonctions à connaitre terminé. Comme nous vous l'avons déjà dit, si vous voulez pousser plus loin votre connaissance de l'Atari, il vous sera nécessaire d'acquerir un livre traitant en détails du système d'exploitation de l'Atari.

4.3.5. ANNEXE : FONCTIONS MINITEL

Ce programme comprend toutes les fonctions utiles à la gestion du minitel, il ne fait rien en soi mais chaque procédure pourra être utilisée dans vos propres programmes. Ainsi, sauvé en Ascii (grace à SAVE ,A) il pourra etre 'mergé' a tous vos programmes ayant un rapport avec le minitel.

```
END
' Configuration de la sortie Serie pour le minitel
DEF PROC Config
      XBIOS(,$F,7,0,%10101110,1,1,0)
RETURN
' Procedure permettant d'envoyer une chaine vers le minitel
DEF PROC Outstring(Dev,S$)
      LOCAL I
      FOR I=1 TO LEN(S$)
            Out(Dev, ASC( MID$(S$,I,1)))
      NEXT I
RETURN
' Fonction retournant l'état de la Rs232 en entrée
DEF FN Inpstat(Dev)
      LOCAL A
      BIOS (A,1,Dev)
RETURN A
' Fonction permettant de prendre un caractère venant du minitel
DEF FN Inp(Dev)
      LOCAL A
      BIOS (A,2,Dev)
      IF Dev=3 THEN A=A+256
RETURN A
' Procédure permettant d'envoyer un caractère vers le minitel
DEF PROC Out(Dev,A)
      BIOS (,3,Dev,A)
RETURN
' Procédure permettant de vider le tampon d'entrée
DEF PROC Vide(Dev)
      WHILE FN Inpstat(Dev)
            A=FN Inp(Dev)
      WEND
RETURN
```

5. LE GEM

5.1. INTRODUCTION : QU'EST-CE-QUE LE GEM ?

Le Gem est une partie du système d'exploitation de l'Atari St, c'est lui qui va gérer l'environnement graphique, c'est à dire la souris, les fenêtres, les boites de dialogues, les menus déroulants, mais aussi les fonctions graphiques elles mêmes.
Le Gem est l'ensemble de routines qui rend la communication avec votre ordinateur agréable. D'ailleurs, Gem veut dire "Graphic Environnement Manager" ce qui veut dire "gestionnaire de l'environnement graphique". Comme nous allons le voir, le Gem est divisé en deux parties :

- Le Vdi : Dans cette partie sont incluses toutes les fonctions graphiques (comme par exemple, faire un cercle, une droite, remplir une surface)

- L'Aes : Qui contient toutes les fonctions capables de gérer les fenêtres, les menus déroulants, etc...

Les deux chapitres qui vont suivre n'ont pas pour but de vous apprendre tout sur le Gem, loin de la (il existe des ouvrages spécialisés traitant uniquement du Gem) mais de vous familiariser un peut avec cet outil par le biais d'exemples précis, ainsi vous serez capable, si vous le désirez, de maitriser le Gem en vous référant au manuel de programmation.

Bien sûr, toutes les fonctions Gem ne sont pas comprises dans l'Omikron, cela prendrait trop de place pour quelque chose dont tout le monde ne se sert pas. Pour avoir accès a toutes ces fonctions il va donc falloir charger à la suite de votre programme une 'Library' qui s'appelle GEMLIB et qui contient toutes les fonctions Vdi et Aes, c'est à dire toutes les fonctions Gem. Pour charger cette bibliothèque il faut taper au début de votre programme la ligne:

LIBRARY GEM,"A:\GEM.LIB" 'Remplacez le "A:\GEM.LIB" en fonction de l'endroit où se trouve le fichier GEM.LIB".

Automatiquement, le basic ajoutera la ligne 'LIBRARY CODE Gem' en fin de programme. Dès lors, vous pourrez utiliser toutes les possibilités de GEM.

5.1.1. LE VDI

Le VDI (Virtual Device Interface) est la base graphique de votre ordinateur, c'est lui qui sait comment dessiner une droite, un cercle etc...

A l'origine, les fonctions Vdi utilisent pour prendre et retourner des paramètres, cinq tableaux (Contrl, Intin, Ptsin, Intout et Ptsout), mais en Basic Omikron ces tableaux seront tous gérés par la bibliothèque GEMLIB.

En effet, grâce à cette bibliothèque, vous allez pouvoir appeler n'importe quelle fonction GEM exactement comme une procédure en Omikron (d'ailleurs c'est bien simple, les fonction GEM sont définies sous forme de procédures dans GEMLIB !), voici tout de suite un exemple :

Essayons d'appeler la fonction V_Circle du VDI: cette fonction a pour paramètres les coordonnées X et Y du centre du cercle à tracer et le rayon de ce cercle. Voici donc un appel à cette fonction: `V_Circle(X,Y,Rayon)`

Vous voyez que l'appel aux fonction VDI est très simple. Bien entendu, ici il était plus simple de se servir de la fonction Omikron CIRCLE mais il arrive que ce que vous vouliez faire ne corresponde pas exactement à une fonction Omikron. Dans ce cas, une fonction VDI pourra peut-être faire votre bonheur.

Bien entendu, nous n'allons pas mettre la liste exhaustive des fonctions VDI dans ce livre, cela prendrait trop de place et cette liste existe déjà dans tous les livres consacrés au GEM et surtout dans le Manuel de Programmation (Qui sera un de vos plus précieux outils si vous désirez programmer le GEM). Pour mieux vous faire comprendre certains appels à des fonctions VDI, voici un petit programme, reprenant quelques fonctions :

```
' ----- Programme de démonstration du VDI -----
LIBRARY Gem ,"a:\gem.lib"'  Chargement de la bibliothèque GEM
Appl_Init'                  Initialisation de l'application GEM
V_Opnvwk'                   Initialisation pour le graphisme VDI
CLS'                        On efface l'écran
V_Rvon'                     On passe en mode inversé
PRINT @(0,33);"Cliquez !"'  On affiche le texte
V_Rvoff'                    On passe en mode non-inversé
REPEAT
      Vq_Mouse(Bouton,X,Y)' On prend l'état de la souris
UNTIL Bouton<>0'            On attends un click souris
V_Hide_C'                   On efface la souris
V_Circle(150,190,100)'      On affiche le cercle plein
Vsf_Interior(0)'            On choisit le mode vide
V_Circle(490,190,100)'      On affiche le cercle vide
Vsf_Interior(3)'            On choisit le mode hachures
FOR I=1 TO 12
   Vsf_Style(I)'                  On choisit le type de hachures
   V_Circle(25+I*45,50,20)'       On affiche le cercle
NEXT I
Vsf_Interior(2)'                  On choisit le style Motif
FOR I=1 TO 12
   Vsf_Style(I)'                  On choisit le type de motif
   V_Circle(25+I*45,330,20)'      On affiche le cercle du haut
   Vsf_Style(I+12)'               On choisit le type de motif
   V_Circle(25+I*45,375,20)'      On affiche le cercle du bas
NEXT I
REPEAT UNTIL INKEY$ <>""'         On attends l'appui d'une touche
V_Clsvwk'                         On ferme la station de travail VDI
Appl_Exit'                        On ferme l'Appliation GEM
```

Dans n'importe quel programme utilisant Gem, il est nécessaire d'ouvrir l'application. Ceci revient en fait à dire au Gem : ici commence un programme qui va utiliser certaines fonctions du Gem. Cette opération est réalisé par la fonction Gem AES:

`Appl_Init`

Ensuite, il faut initialiser le VDI (seulement si vous voulez l'utiliser!). Pour ce faire, on tape:

`V_Opnvwk`

Pour afficher le texte 'Cliquez!' en inverse vidéo, on utilise les instruction V_Rvon et V_Rvoff qui servent respectivement à passer en vidéo inverse et à revenir en mode normal.

Ensuite on attend un click souris en utilisant la fonction: `Vq_Mouse(Bouton,X,Y)`

Qui va nous retourner l'état du bouton dans la variable Omikron Bouton, l'abcisse de la souris dans la variable X et son ordonnée dans la variable Y. (Dans le cas présent, seul la variable Bouton nous servira)

Il va bien sur falloir cacher la souris avant les instructions graphiques, nous allons donc utiliser l'instruction Gem:

`V_Hide_C`
(La fonction inverse, qui réaffiche la souris, est V_Show_C).

Par la suite, nous allons afficher différents cercles, grace bien sur à l'instruction V_circle (Que nous avons vu precedement), mais aussi grace aux instructions Vsf_interior et Vsf_style dont voici la syntaxe:

`Vsf_interior(mode)`

Cette instruction sert donc a changer le mode de remplissage, c'est l'équivalent d'une partie de FILL STYLE= en Omikron. Voici les différentes valeurs que peut prendre <mode> :

0 : Vide
1 : Noir
2 : Motif
3 : Hachures
4 : Motif défini par l'utilisateur

`Vsf_Style(style)`

Celle ci sert a choisir le remplissage (c'est à dire, avec les modes 2 et 3, le type de motif ou le type de hachures), <Style> peut prendre des valeurs de 1 à 24 pour les motifs (mode 2) et de 1 à 12 pour les hachures (Mode 3)

Et enfin, à la fin de notre programme, il va falloir fermer l'application et le système graphique VDI, ceci se fait grace aux instructions inverses de Appl_init et V_opnwvk, c'est à dire :

```
V_clsvwk
Appl_exit
```

Ceci sert a montrer au Gem que le programme est terminé.

Comme vous avez du le remarquer, la programmation du VDI est souvent assez lourde, de plus, il existe la plupart du temps des équivalents Omikron qui sont plus faciles à utiliser. toutefois, il est toujours bon de se pencher un minimum sur la façon dont travaille l'ordinateur, de plus, certaines fonctions VDI n'ont pas d'équivalent direct en Basic Omikron.

5.1.2. L'AES

L'AES (Application Environement System) est la partie de GEM la plus complexe. C'est elle qui permet de contrôle des objets graphiques tels que les fenêtres, les boites de dialogues, les menus déroulants, les icônes, etc... Pour ce faire, l'AES utilise des fonctions du VDI.

Comme pour le VDI, nous n'expliqueront pas l'intégralité de l'AES, ce qui prendrait trop de place, et qui ne serait que repeter certains ouvrages sur le GEM. Nous nous contenterons d'un programme minimum, qui vous montrera comment démarrer avec l'AES. Les fonctions de l'AES, tout comme celles du VDI sont toutes incluses dans GEMLIB, ainsi pour la procédure de chargement de cette librairie, referez vous au chapitre VDI.

```
' ------ Démonstration de quelques fonctions AES -----
LIBRARY Gem ,"A:\GEM.LIB"'    Chargement de la bibliothèque GEM
Appl_Init'                     Initialisation de l'AES
V_Opnvwk'                      Initialisation du VDI
V_Clrwk'                       Effacement de l'écran
Graf_Mouse(0)'                 Choix d'un motif pour la souris
MOUSEON '                      souris allumée
' - on passe en revue les différents pointeurs de la souris...
FOR K=1 TO 7
      Graf_Mouse(K)'           Changement du pointeur
      WAIT 2'                  Attente de deux secondes
NEXT K
' ---- Affichage d'un cadre d'alerte
FORM_ALERT (1,"[1][Texte dans la boite][OK|NON]",Button)
IF Button=1 THEN PRINT @(0,30);"Choix: OK"
IF Button=2 THEN PRINT @(0,30);"Choix: NON"
'---- on change de cadre d'alerte
FORM_ALERT (1,"[2][Texte sur|deux lignes][OK|NON]",Button)
' --- la meme avec l'autre bouton par défaut et différent logo
FORM_ALERT (2,"[3][Text sur|deux lignes][OK|NON]"
```

S. ENSELME

PROGRAMMER FM MELODY MAKER (III)

Il ne fait aucun doute qu'après les deux précédents articles, vous maîtrisez parfaitement la programmation de la cartouche FM, n'est-ce pas ? Vous avez réalisé des dizaines de sons, et n'attendez plus que le moyen d'utiliser ces nouveaux instruments. C'est pourquoi le sujet de ce dernier épisode est la récupération des morceaux réalisés sous le Melody Maker, avec le format de ses fichiers.

EN-TETE

Toutes les musiques que vous sauvez en .MUS à partir du Melody Maker (FM ou STE) respectent le même format, à savoir 96 octets d'en-tête suivis des données. Cet en-tête n'est malheureusement pas décrite dans la documentation développeur du FM Melody Maker, les auteurs spécifiant qu'elle est sans intérêt pour les utilisateurs... Armés de patience et de deux ou trois outils, nous avons tout fait pour en comprendre le contenu. Nous vous livrons dans le premier tableau le résultat de cette recherche, soit 75 octets, les 21 autres ne nous ayant pas livré leur secret...

DONNÉES

Une fois ces octets étudiés, nous pouvons passer aux données du morceau. Elles suivent immédiatement l'en-tête, et sont formées de couples d'octets. Le premier décrit le type et le second son paramètre. Reportez-vous alors au second tableau.
Enfin, le troisième tableau donne la correspondance entre le paramètre "Longueur" et la durée effective.

EXEMPLES

Pour plus de clarté, voici quelques exemples de notes :

```
3C,02 correspond à une noire de hauteur do3.
3C,08   "        "  noire pointée   "
3C,43   "        "  croche avec une liaison.
```

C'est donc avec ces fichiers que nous concluons cette brève série, en espérant qu'elle vous a permis d'aller plus loin avec votre FM Melody Maker. A présent, nous vous souhaitons une bonne programmation...

S. MOUGEY

"LES DONNÉES"

Val	Objet	Oct suivant
0	Silence	Longueur
C à 68	Note Midi	Longueur
80	Accord	N° d'accord
81	Repet. Début	N°
82	Repet. Fin	N°
83	Repet. Term	N° Term
84	Repet. DC	N°
85	Repet. DS	N°
86	Repet. Coda	N°
87	Repet. Fine	N°
88	Repet. Segno	N°
89	Aller à la Coda	N°
90	Armure	N° Armure
91	Signature	N° Signat.

"LES DURÉES"

Val.	Longueur
0	Ronde
1	Blanche
2	Noire
3	Croche
4	Double Croche
5	Triple Croche
+6	Pointée
+C	Triolet
+40	Liaison

"FORMAT DE L'EN-TÊTE"

Oct	Paramètre	De	à	Oct	Paramètre	De	à	Oct	Paramètre	De	à
4	Canal Pied	0	F	1A	Vélocité Pied	1	7F	36	Velocité Acc. 4	1	7F
5	Canal Caisse	0	F	1B	Vélocité Caisse	1	7F	37	Velocité Basse	1	7F
6	Canal HiHat	0	F	1C	Vélocité HiHat	1	7F				
7	Canal Cymbales	0	F	1D	Vélocité Cymbales	1	7F	38/39	Tempo	28	F0
8	Canal TomTom	0	F	1E	Vélocité TomTom	1	7F				
9	Canal Midi A	0	F	1F	Vélocité Midi A	1	7F	3C/3D	Accompagnement	0	F
A	Canal Midi B	0	F	20	Vélocité Midi B	1	7F				
B	Canal Midi C	0	F	21	Vélocité Midi C	1	7F	3E/3F	Mixage Mélodie	0	F
C	Canal Midi D	0	F	22	Vélocité Midi D	1	7F	40/41	Mixage Acc. 1	0	F
D	Canal Midi E	0	F	23	Vélocité Midi E	1	7F	42/43	Mixage Acc. 2	0	F
E	Canal Midi F	0	F	24	Vélocité Midi F	1	7F	44/45	Mixage Acc. 3	0	F
								46/47	Mixage Acc. 4	0	F
F	Note Pied	1	7F	2C	Canal Mélodie	0	F	48/49	Mixage Basse	0	F
10	Note Caisse	1	7F	2D	Vélocité Mélodie	1	7F	4A/4B	Mixage Rythmique	0	F
11	Note HiHat	1	7F								
12	Note Cymbales	1	7F	2E	Canal Acc. 1	0	F	4C/4D	N'Instr Mélodie	1	4E
13	Note TomTom	1	7F	2F	Canal Acc. 2	0	F	4E/4F	N'Instr Acc. 1	1	F
14	Note Midi A	1	7F	30	Canal Acc. 3	0	F	50/51	N'Instr Acc. 2	1	F
15	Note Midi B	1	7F	31	Canal Acc. 4	0	F	52/53	N'Instr Acc. 3	1	F
16	Note Midi C	1	7F	32	Canal Basse	0	F	54/55	N'Instr Acc. 4	1	F
17	Note Midi D	1	7F	33	Velocité Acc. 1	1	7F	56/57	N'Instr Basse	1	F
18	Note Midi E	1	7F	34	Velocité Acc. 2	1	7F				
19	Note Midi F	1	7F	35	Velocité Acc. 3	1	7F				

INITIATION AU BASIC 1000D (II)

Le Basic 1000D, nous l'avons vu, permet de gérer d'une part des calculs numériques en utilisant un grand nombre de chiffres, et d'autre part de travailler sur des calculs littéraux, en permettant notamment une simplification de fractions rationnelles. Nous allons aujourd'hui utiliser ces deux particularités pour étudier quelques propriétés de certains polynômes.

L'ÉTAT DE DÉCOMPOSITION EST AVANCÉ

Le polynôme $p(x) = x^3 + x^2 + x + 1$ est un polynôme assez remarquable. Une identité classique permet d'écrire :

$x^4 - 1 = (x - 1).p(x)$.

C'est-à-dire que ses racines sont les racines quatrièmes de l'unité, sauf 1. P(x) peut donc se décomposer sur **R** en $(x + 1)(x^2 + 1)$. Sur C, on peut écrire :

$$p(x) = (x + 1)(x + i)(x - i).$$

On retrouve la propriété connue : dans **C**, tout polynôme de degré n admet n racines distinctes ou non. Ici, ces racines sont -1 (on retrouve la seule racine réelle) et i et -i, racines complexes. Rappelons que les nombres i et -i vérifient $i^2 = -1$, $i^3 = -i$ et $i^4 = 1$, ce qui permet de connaître toutes les puissances de i... Nous en aurons besoin sous peu.

La conjecture que nous allons étudier est celle-ci :

Le polynôme $x^{3n} + x^{2n} + x^n + 1$ est divisible par $x^3 + x^2 + x + 1$ si et seulement si n est impair.

Nous avons vu que p(x) admet pour seules racines complexes -1, i et -i. Soit $f(x) = x^{3n} + x^{2n} + x^n + 1$.

Pour montrer que f(x) est divisible par p(x), il suffit de montrer que -1, i et -i sont racines de f(x), puisque ces racines sont racines simples de p(x). Comme $x^{3n} = (x^3)^n$, etc, on peut écrire :

$$f(-1) = (-1)^n + 1^n + (-1)^n + 1 = 2(-1)^n + 2$$

Donc f(x) n'est divisible par (x + 1) que si n est impair.Sinon f(-1) prend pour valeur 4.
Montrons alors que i et -i sont aussi racines de f(x) :

$$f(i) = (-i)^n + (-1)^n + i^n + 1$$

Si n est multiple de 4, f(i) = 1+1+1+1 = 4
Si n est multiple de 4 plus 1 (donc impair), f(i) = -i-i+i+1 = 0
Si n est multiple de 4 plus 2, f(i) = -1+1-1+1 = 0, donc f(x) est divisible par (x - i). (Mais pas par x - 1 !).
Si n est multiple de 3 plus 1 (donc impair), f(i) = i-1-i+1 = 0

Un calcul semblable montre que f(-i) prend dans chacun des cas les valeurs 4, 0, 0, 0 .

Si l'on ne connaît pas les nombres complexes, on peut remarquer que f(x) peut s'écrire $f(x) = (x^n+1)(x^{2n}+1)$. Comme $(-1)^n$ vaut -1 si n est impair, (-1) est racine de $x^n + 1$, pour n impair. Il existe donc un polynôme q(x) tel que $(x^n+1) = (x+1).q(x)$. On en déduit, en remplaçant x par x^2, que $(x^{2n}+1) = (x^2+1).q(x^2)$, puis $f(x) = (x+1)(x^2+1).q(x).q(x^2)$, donc $f(x) = p(x).q(x).q(x^2)$.

DEMANDEZ LE PROGRAMME !

Voici un petit programme en Basic 1000D qui permet de vérifier cette propriété : si n est pair, le résultat donne le

quotient de deux polynômes. Sinon, un seul polynôme est affiché, le polynôme quotient.

```
for n=1 to 51
 print n;" : "
 print ((x^3)^n+(x^2)^n+ x^n+ 1.) / (x^3+ x^2+ x+ 1.)
 print
 w$=input$(1)
next n
```

Voici les décompositions obtenues pour quelques valeurs impaires de n :

```
1 : 1
3 : x^6 - x^5 + x^3 - x + 1.
5 : x^12 - x^11 + x^8 - x^6 + x^4 - x + 1.
7 : x^18 - x^17 + x^14 - x^13 + x^11 - x^9 + x^7 - x^5
    + x^4 - x + 1.
9 : x^24 - x^23 + x^20 - x^19 + x^16 - x^14 + x^12
    - x^10 + x^8 - x^5 + x^4 - x + 1.
17 : x^48 - x^47 + x^44 - x^43 + x^40 - x^39 + x^36
     - x^35 + x^32 - x^30 + x^28 - x^26 + x^24 - x^22
     + x^20 - x^18 + x^16 - x^13 + x^12 - x^9 + x^8
     - x^5 + x^4 - x + 1.
23 : x^66 - x^65 + x^62 - x^61 + x^58 - x^57 + x^54
     - x^53 + x^50 - x^49 + x^46 - x^45 + x^43 - x^41
     + x^39 - x^37 + x^35 - x^33 + x^31 - x^29 + x^27
     - x^25 + x^23 - x^21 + x^20 - x^17 + x^16 - x^13
     + x^12 - x^9 + x^8 - x^5 + x^4 - x + 1.
31 : x^90 - x^89 + x^86 - x^85 + x^82 - x^81 + x^78
     - x^77 + x^74 - x^73 + x^70 - x^69 + x^66 - x^65
     + x^62 - x^61 + x^59 - x^57 + x^55 - x^53 + x^51
     - x^49 + x^47 - x^45 + x^43 - x^41 + x^39 - x^37
     + x^35 - x^33 + x^31 - x^29 + x^28 - x^25 + x^24
     - x^21 + x^20 - x^17 + x^16 - x^13 + x^12 - x^9
     + x^8 - x^5 + x^4 - x + 1.
49 : x^144 - x^143 + x^140 - x^139 + x^136 - x^135
     + x^132 - x^131 + x^128 - x^127 + x^124 - x^123
     + x^120 - x^119 + x^116 - x^115 + x^112 - x^111
     + x^108 - x^107 + x^104 - x^103 + x^100 - x^99
     + x^96 - x^94 + x^92 - x^90 + x^88 - x^86 + x^84
     - x^82 + x^80 - x^78 + x^76 - x^74 + x^72 - x^70
     + x^68 - x^66 + x^64 - x^62 + x^60 - x^58 + x^56
     - x^54 + x^52 - x^50 + x^48 - x^45 + x^44 - x^41
     + x^40 - x^37 + x^36 - x^33 + x^32 - x^29 + x^28
     - x^25 + x^24 - x^21 + x^20 - x^17 + x^16 - x^13
     + x^12 - x^9 + x^8 - x^5 + x^4 - x + 1.
```

Il est intéressant de voir comment se répartissent les coefficients des différentes puissances de x. Le premier coefficient non nul correspond - évidemment - à la puissance 3n-3, puisque l'on divise un polynôme de degré 3n par un polynôme de degré 3. On peut remarquer que les coefficients sont alternativement égaux à 1 et à -1, mais que certaines puissances de x seulement en sont affectées.
Au début, la règle est la suivante : on conserve deux puissances consécutives, on en élimine deux, et ainsi de suite. Par contre, lorsque la puissance est comprise entre n et 2n, on ne conserve qu'une puissance de x sur deux. Entre n et 0, on retrouve la première règle.

LES 3 DANS LES GRANDES LARGEURS

Si l'on pose x = 3 dans la décomposition précédente, on trouve : 3^3n + 3^2n + 3^n + 1, qui est divisible par 3^3 + 3^2 + 3 + 1, c'est-à-dire la propriété suivante :

Le nombre 27^n + 9^n + 3^n + 1 est divisible par 40 si et seulement si n est impair.

Le petit programme en Basic 1000D suivant permet de calculer le nombre *27^n + 9^n + 3^n + 1* en fonction de n. Le pouvoir numérique du Basic 1000D permet de calculer ce nombre de manière exacte, même s'il comporte plus de sept cents chiffres !

```
for n=1 to 500
 a=27^n+9^n+3^n+ 1.
 print n;" : ";
 print a
 print (a/40)-int(a/40)
 print
 w$= input$(1)
next n
```

Voici les valeurs de 27^n + 9^n + 3^n + 1 pour différentes valeurs de n impair. On peut constater que chacun de ces nombres est divisible par 40, le nombre formé par ses trois derniers chiffres étant un multiple de 40. Pour les nombres pairs, le programme donne le reste exprimé sous forme décimale. Il est alternativement égal à 1/10 ou 1/2 (puisque une fois sur deux, le polynôme est divisible par (x^2 + 1), c'est-à-dire 10, sans être divisible par (x + 1), soit 4.

```
 1 :  40
 3 :  20440
 5 :  14408200
 7 :  10465138360
 9 :  7625984925160
11 :  5559091947792280
13 :  4052557694886398920
15 :  2954312912441980142200
17 :  2153693979752739595117480
19 :  1570042900432933330475788120
21 :  1144561273540256484027922408840
23 :  834385168339943471891603972970040
25 :  608266787714075607107376692496241000
27 :  443426488243095919685252678304437767960
29 :  323257909929179244420971227034825570821960
31 :  235655016338368617019492208892320940227301880
33 :  171792506910670474581974759221158345215244199720
```

et voici quelques grandes valeurs à titre d'exemple (la moitié du magazine ne suffirait pas à vous en fournir la liste exhaustive !) :

```
51 : 9989689095948428268966925764593495981136753038674841
     877564508819881211480

123 : 114220219797390847899906202998097281809766047839867
      909377929082803879290950616227359692726655890776742
      188896100876189871050950324698043980057075861515
      163546945463448745973718040

333 : 4406902731602688789634850865840481219884740109173
      8272255497345607560953244890163318025943795020268
      7321303259232290860785316984860700206303955114241
      7526512246764525111707907083553895039965611642701
      8113466392642569933534956243005760143148588669341
      3524110535314550440272152965153222164372499812247
      6958343699384156082282958040918448726421063934346
      3179718194144948589777409450116764572443366930367
      9706631740966181990341357894673130566555535118900
      978685396866390028547447981865611720

499 : 1780402991523222882719200002365200921971942490552
      7598109605371747222645665526590873592287505858695
      2284158460651657698483729552422311747645060479906
      7314196069667730019850858406651711841925611882611
      6553472137694309928848405341808275335527334981894
```

```
1118741298175151001407975775029196968887396065355
4444273233279864595778244809196525193481630568316
0104459966726812165396014038822512035450145167842
9753190554901579003181150688554084645000100450588
3179573825097886143639236037852398621686369837979
6639665716102139774536269814704036498520634709237
6433874291981780595891748418692241599667321247162
5220081531158220589838936877133198081226778119648
5964517852483706380058730478956830310729979219353
38187225613451834562570258520
```

Le calcul exact de tels nombres demanderait d'énormes efforts en Basic classique. Il prend quelques dizaines de secondes pour l'ensemble avec le Basic 1000D.

Notez que le choix du nombre 3 est tout à fait arbitraire. On pourrait montrer aussi bien que, toujours à la condition que n soit impair, :

8^n + 4^n + 2^n + 1 est divisible par 15,
125^n + 25^n + 5^n + 1 est divisible par 156,
1000^n + 100^n + 10^n + 1 est divisible par 1111,

etc...

En fait, ce problème est un cas particulier, pour k = 3, de la propriété suivante :

Le polynôme 1 + x + x^2 + x^3 + ... + x^k est un diviseur du polynôme 1 + x^n + x^2n + x^3n + ... + x^kn si n et k+1 ont un pgcd égal à 1.

Bon courage pour vérifier cette assertion pour n et k quelconques, mais vous pouvez toujours vous faire aider par le Basic 1000D !

Jean Pascal DUCLOS

INITIATION AU BASIC GFA - XI (Série n° 2)

CLAVIER, CLAVIER, SAIS-TU VRAIMENT QUI JE SUIS ?

Votre information ne serait pas complète, si vous ne saviez pas exploiter parfaitement toutes les ressources de votre clavier. Cet organe d'échange, entre vos programmes et l'utilisateur, est d'une importance capitale. Pour gérer correctement votre clavier, vous devrez manipuler des fonctions choisies parmi un vaste ensemble ; vous verrez que le vrai problème réside dans ce qu'il est convenu d'appeler "l'embarras du choix". Nous allons donc nous attacher à préciser les options possibles dans la *deuxième* partie de cet article.
En effet, dans un premier temps, nous ferons un nécessaire détour dans la direction de la conversion de données. Les notions qui y seront présentées, trouveront très naturellement leur application dans la gestion du clavier.

CONVERSION DE DONNÉES

Comme vous le savez, la seule information "physiquement" présente dans l'ordinateur est de type binaire. Pour pouvoir exploiter ces informations, il faut pouvoir les convertir, c'est-à-dire les traduire sous une forme qui les rende intelligibles. C'est ainsi que nous pourrons convertir une donnée en donnée numérique, ou en chaîne de caractères par exemple. La conversion permet finalement de passer une donnée d'un type dans un autre.

Des conversions de données ont été utilisées maintes fois dans nos programmes : vous souvenez-vous des fonctions ASC, BIN$, CHR$, et VAL ? Voici un bref rappel :

- ASC(car) : retourne le code ASCII du caractère 'car' ;
- BIN$(val,n_bits) : retourne une chaîne binaire (de longueur égale à 'n_bits'), équivalente à la variable entière 'val' ;

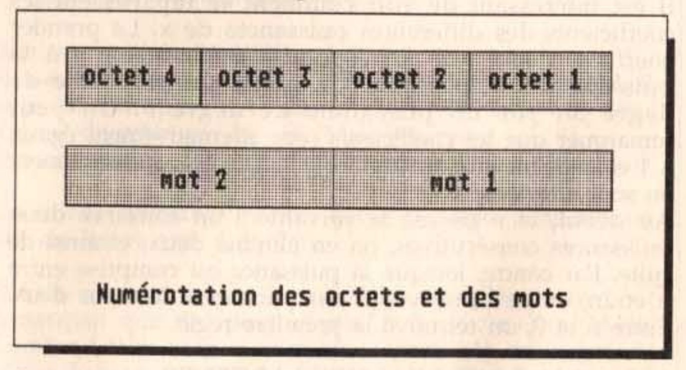
- CHR$(asc) : effectue le contraire de ASC (retourne le caractère dont on transmet le code ASCII) ;

- VAL(chaîne) : convertit une chaîne de caractères numériques dans la valeur numérique correspondante.

Les fonctions MKI$ et MKL$: ces fonctions effectuent une conversion quelque peu artificielle, mais souvent très pratique. Il s'agit, en fait, d'un petit tour de passe-passe, qui consiste à placer un mot court (ou un mot long) dans une chaîne de caractères. En général, l'objectif n'est pas de manipuler la chaîne ainsi créée comme une "vraie" chaîne de caractères (pour l'éditer, par exemple), mais de l'utiliser comme un "réservoir de stockage" (ou un tampon). La syntaxe de ces fonctions est la suivante :

chaîne$=mki$(mot_court)
chaîne$=mkl$(mot_long)

On transmet un mot court (16 bits) à MKI$ et un mot long (32 bits) à MKL$. Vous retrouvez la lettre "I", dans MKI$, pour signifier "Integer" et "L", dans MKL$, qui signifie "Long". En général, on place un mot dans une chaîne de caractères, pour accéder à l'un des octets qui constitue ce mot. Qu'est-ce à dire ? Formulons brutalement le problème posé :



Numérotation des octets et des mots

Considérez notre schéma : comment feriez-vous pour récupérer l'octet numéro 2 (de l'un ou l'autre des deux mots représentés) ? Il existe beaucoup de solutions pour y parvenir, plus ou moins directes et compréhensibles. Nous avons choisi la méthode, qui consiste à placer le mot dans une chaîne, pour pouvoir *utiliser des fonctions de manipulation de chaînes de caractères*. En effet, les fonctions LEFT$, MID$ et RIGHT$ permettent d'extraire

un caractère (c'est-à-dire un octet !) d'une chaîne de caractères. Nous allons voir comment.

Les cinq lignes de programme suivantes effectuent la tâche que nous nous sommes assignés, c'est-à-dire accéder à chaque octet qui compose un mot (en l'occurrence, un mot long) :

```
quatre_car$=MKL$(mot_long%)
octet_1|=ASC(RIGHT$(quatre_car$))
octet_2|=ASC(MID$(quatre_car$,3,1))
octet_3|=ASC(MID$(quatre_car$,2,1))
octet_4|=ASC(LEFT$(quatre_car$))
```

Le principe est de placer le mot long dans une chaîne de quatre caractères, puis d'extraire chaque caractère avant de le convertir en un octet (grâce à ASC).

Remarque : dans cet exemple, et dans tout ce qui va suivre, ne perdez pas de vue la convention, suivant laquelle on numérote les octets et les mots *en partant de la droite* (suivant cette convention, l'octet ou le mot '1' figurent à droite, voyez notre schéma).

Nos cinq lignes de programme sont aisément compréhensibles, et compte tenu de la généralité du problème posé, il est intéressant de les intégrer dans une procédure comme celle-ci :

```
PROCEDURE split_long(mot_long%)
  LOCAL quatre_car$
  '                        Rappel : octets = 4 3 2 1
  '                                 mots   =  2   1
  quatre_car$=MKL$(mot_long%)
  octet_1|=ASC(RIGHT$(quatre_car$))
  octet_2|=ASC(MID$(quatre_car$,3,1))
  octet_3|=ASC(MID$(quatre_car$,2,1))
  octet_4|=ASC(LEFT$(quatre_car$))
  mot_1&=octet_1|+(256*octet_2|)
  mot_2&=octet_3|+(256*octet_4|)
RETURN
```

Par souci de simplicité, cette procédure fait appel à six variables globales, qui contiennent le résultat de la conversion. Les deux dernières lignes (avant RETURN) permettent d'accéder aux mots correspondants au schéma précédent : le mot long est "divisé" en ses deux constituants de base, c'est-à-dire deux mots courts.

Les vrais anglophiles - qui nous font l'honneur de nous lire - n'ignorent pas que le terme "split" signifie la scission (d'une entité en ses divers composants). Voilà qui définit parfaitement notre objectif de départ, et nous conduit à ce petit programme montrant les vertus de notre procédure récemment créée (insérez-la à la suite de ce programme) :

```
'                    SPLIT.GFA
mot_long%=1145789
split_long(mot_long%)
PRINT "Mot long = ";mot_long%,BIN$(mot_long%,32)
PRINT "Mot    1 = ";mot_1&,BIN$(mot_1&,16)
PRINT "Mot    2 = ";mot_2&,BIN$(mot_2&,16)
PRINT "Octet 1  = ";octet_1|,BIN$(octet_1|,8)
PRINT "Octet 2  = ";octet_2|,BIN$(octet_2|,8)
PRINT "Octet 3  = ";octet_3|,BIN$(octet_3|,8)
PRINT "Octet 4  = ";octet_3|,BIN$(octet_4|,8)
```

Ce programme n'a pas d'autre volonté que d'être démonstratif : il "splite", en octets et en mots courts, un mot long. Ce mot long est arbitrairement choisi, et vous êtes invité à introduire toute autre valeur plus conforme à votre goût personnel.

Afin que vous puissiez vérifier *de visu* les résultats, nous avons employé la fonction BIN$, qui édite sous forme binaire chacun des éléments. Vous constaterez que les données de base ne sont absolument pas altérées par nos manipulations, mais qu'elles se trouvent uniquement décomposées.

Transmettre un mot court : vous avez la possibilité de transmettre un mot court (16 bits) à notre procédure pour la décomposer en deux octets (octets 1 et 2). Bien évidemment, le mot 2 (mot_2&) et les octets 3 et 4 (octet_3| et octet_4|) seront nuls.

Les fonctions BYTE, CARD : spécifiques à GfA 3, ces fonctions acceptent un mot long comme argument. BYTE retourne l'octet inférieur (octet 1) ; CARD retourne le mot inférieur (mot 1) du mot long transmis à la fonction. Dans certains cas, vous pourrez donc les employer à la place de la procédure 'split_long'.

LA GESTION DU CLAVIER

Le fait de pouvoir décomposer les mots en octets va nous ouvrir de nombreuses portes, et notamment celles de la gestion du clavier. En effet, certaines fonctions clavier retournent un mot ou un mot long, dont il faut extraire un certain nombre d'informations.
Avant d'entrer dans les détails, nous allons préciser les deux grands modes de gestion du clavier (mode scrutation et mode attente).

Scrutation ou attente ?

Il y a deux façons de tester l'état du clavier : soit vous attendez que l'utilisateur frappe une touche, soit vous créez une boucle de scrutation pour voir si l'état du clavier est altéré.

Si le distinguo entre les deux modes ne vous apparaît pas clairement, voici une petite analogie. Vous attendez une carte postale. Il y a deux façons de faire : soit vous campez devant la boîte à lettres (mode *attente*), soit vous passez de temps en temps jeter un petit oeil sur la boîte (mode *scrutation*), tout en vaquant à vos occupations familières. Dans la réalité, le second mode serait le plus adapté, mais dans le cadre de la gestion de clavier, les deux modes ont leurs avantages.

Le mode "attente" : le mode attente ne vous est pas inconnu, puisque nous avons souvent fait usage de deux fonctions ayant cette caractéristique : les fonctions BIOS(2,2) et INP(2). Ces fonctions attendent que l'utilisateur se décide à presser gentiment une touche ; s'il ne le fait pas, le programme va l'attendre indéfiniment (prévoir un numéro vert et de la nourriture en quantité suffisante).

La solution qui consiste à attendre qu'une touche soit frappée, a l'avantage d'être simple pour le programmeur, mais elle n'est pas toujours parfaitement explicite pour l'utilisateur. Un état d'attente doit *toujours être signalé*, afin que l'utilisateur sache ce que le programme attend de lui. La véritable ambiguïté vient de cet autre périphérique d'entrée qu'est la souris : l'utilisateur est-il censé choisir une touche ou manipuler la souris ? Ne laissez pas l'utilisateur dans l'expectative !

Remarque : lorsque la solution "scrutation" est employée, il est possible de rectifier une entrée incorrecte. Par exemple, si l'utilisateur est censé faire une entrée au clavier, mais qu'il s'empare de la souris, le programme est à même de lui signaler son erreur. Ce n'est pas le cas dans le mode "attente".

La fonction KEYGET : cette fonction est équivalente à BIOS(2,2), à condition que vous disposiez de GfA 3.

Le mode "scrutation" : venons-en à la seconde façon de gérer des entrées clavier, le mode "scrutation". Si votre mémoire est bonne, vous avez rencontré ce terme dans notre précédent article, où il était question de scruter divers événements ; l'un de ces événements était la pression d'une des touches du clavier (les deux autres événements étaient un événement-menu et un événement-souris).
Rappelons que les deux instructions, qui permettent de réaliser une scrutation du clavier, sont celles-ci :

```
ON MENU KEY GOSUB gere_clavier
.../...
DO
  ON MENU      ! Boucle de scrutation du menu (et du clavier)
LOOP
```

Remarque : précisons un petit point de vocabulaire. Pour tester la pression sur une touche, on construit une boucle qui teste *dynamiquement* l'état du clavier ; en revanche, le mode "attente" réalise une situation de test *statique*. Vous rencontrerez parfois, en lieu et place de "attente" et "scrutation", les termes "test statique" et "test dynamique".

*La fonction INKEY$* : d'autres fonctions permettent de scruter l'état du clavier ; notre préférence s'est portée sur la fonction INKEY$, que voici, insérée dans un programme illustrant ses mérites :

```
'           INKEY1.GFA
DO
  deux_car$=INKEY$
  PRINT deux_car$,LEN(deux_car$)
  PAUSE 10
  PRINT "---"
LOOP
```

Pour sortir de cette boucle de scrutation - en l'occurrence il s'agit d'une boucle infinie - vous presserez les touches CONTROL, SHIFT et ALTERNATE. La fonction INKEY$ retourne une chaîne de caractères ; cette chaîne est de longueur variable, comme vous pouvez le voir grâce au second paramètre édité (la fonction LEN retourne la longueur de la chaîne qu'on lui transmet).

La longueur de la chaîne retournée a une valeur informative : si la longueur est nulle, la fonction ne retourne rien (pas de touche pressée) ; si la longueur est égale à un caractère, ce caractère n'est autre que le code ASCII ; nous verrons ultérieurement à quel cas correspond le retour de deux caractères.

La fonction KEYTEST : cette fonction est spécifique à GfA 3, elle équivaut à INKEY$.

Maintenant que la différence est claire entre le mode "attente" et le mode "scrutation", nous allons introduire un nouveau critère de différenciation des fonctions de gestion du clavier.

Avec ou sans écho ?

L'écho est tout simplement une information de retour destinée à informer l'utilisateur. Cette information est envoyée sur le périphérique de sortie actuellement ouvert (en général, il s'agit de l'écran). En clair, si l'utilisateur frappe la touche 'a', et si votre programme est en mode de saisie *avec* écho, le caractère 'a' va apparaître sur l'écran.

Comme vous le savez, la fonction BIOS(2,2) ne réalise PAS un tel écho, et nous avons dû, dans notre premier exemple, la faire suivre d'une instruction PRINT destinée à éditer le caractère frappé (quand il est effectivement éditable !).

Pourquoi l'écho n'est pas toujours souhaitable : a priori, on serait tenté de penser naïvement, qu'il *faut* toujours qu'il y ait écho, c'est-à-dire édition du caractère frappé par l'utilisateur. Faux : vous souhaitez que l'utilisateur frappe sur une touche numérique, mais ce grand nigaud frappe par erreur sur une lettre : vous n'éditez PAS cette lettre, mais vous émettez un bip ou un message destiné à réveiller l'utilisateur. Autrement dit, l'écho n'est souhaitable que lorsque la saisie s'effectue correctement, il faut parfois le remplacer par un message approprié.

La fonction GEMDOS(1) : certaines fonctions ou instructions génèrent automatiquement un écho ; c'est le cas de la fonction GEMDOS(1), comme le montre ce très simple exemple :

```
'             ECHO.GFA
FOR i%=1 TO 10
  x%=GEMDOS(1)
  PRINT
  PRINT x%
NEXT i%
```

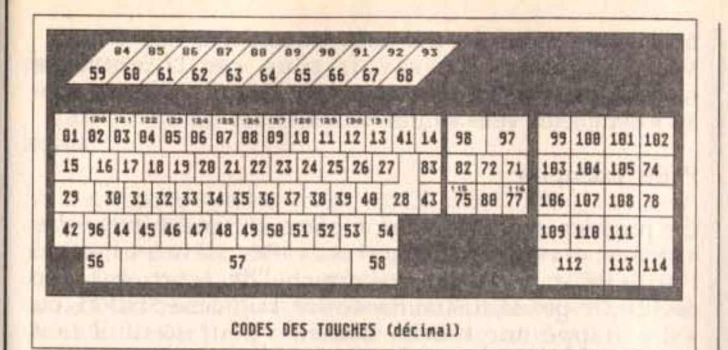
Il est possible de frapper successivement sur dix touches, avant que le programme ne s'arrête. La fonction GEMDOS(1) retourne une valeur entière, permettant de déterminer le code ASCII et le code de la touche frappée (nous allons détailler ultérieurement cette notion) ; elle édite le caractère choisi (écho), lorsqu'il est éditable.
La fonction GEMDOS(1) fonctionne dans le mode "attente" ; on la désigne souvent par l'intermédiaire de l'identificateur: "CCONIN".

Récapitulatif : avant d'aller plus loin, récapitulons. Il existe quatre situations de base : les fonctions d'attente sans écho (exemple : BIOS(2,2) et INP(2)) ; les fonctions d'attente avec écho (exemple : GEMDOS(1)) ; les fonctions de scrutation sans écho (exemple : INKEY$) ; les fonctions de scrutation avec écho (non abordées).
Les conditions sont réunies pour que nous compliquions, encore un peu, tout ce que nous venons d'établir.

Code ASCII, code touche et code complément

Nous allons pouvoir exploiter les notions vues au début de l'article, concernant la décomposition d'un mot en octets. En effet, lorsqu'une touche est frappée, trois informations sont potentiellement disponibles : le code ASCII, le code de la touche, et un code complémentaire (chaque code est placé dans un octet).
Prenons un exemple : vous frappez sur la touche portant le caractère 'a' :

- le code ASCII est bien évidemment celui qui correspond au caractère 'a', c'est-à-dire le code '97'. Vous êtes ici en terrain connu, toutefois, n'hésitez pas à relire le quatrième article de notre série, consacré aux codes ASCII ;
- le code touche, correspondant à la touche 'a', est le code '16'. Reportez-vous à notre schéma de la table des


CODES DES TOUCHES (décimal)

codes clavier, ou codes de touche ;
- le code complémentaire, ou code complément, n'est pas affecté (si vous n'avez pressé QUE la touche 'a').

Le code complémentaire permet de savoir si les touches SHIFT, CONTROL et ALTERNATE ont été pressées.

Comme vous pouviez vous y attendre, nous allons vous proposer un cas un peu plus complexe : l'utilisateur frappe la même touche 'a', ainsi que la touche SHIFT, pour obtenir le 'a' majuscule (A) :

- le code ASCII correspondant sera cette fois '65' (et non '97'), code ASCII du caractère 'A' ;
- le code de touche est '16', comme précédemment (vous avez effectivement frappé la même touche !) ;
- le code complément est égal à 2, puisque la touche SHIFT a été pressée.

Scan code : l'expression anglaise "scan code" est équivalente à "code touche", et vous la rencontrerez souvent dans la littérature (snob).

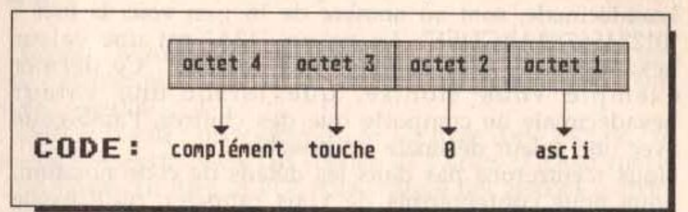
Code ASCII nul : comme vous l'aviez pressenti, toutes les touches ne vont pas générer de code ASCII. En effet, certaines touches sont uniquement définies par leur code touche ; dans ce cas, la "valeur" du code ASCII retourné est '0'. C'est notamment le cas des quatre touches fléchées, des dix touches de fonction, etc.
Afin de pouvoir juger par vous-même, voici un petit programme, qui va illustrer les points que nous avons rencontrés (n'oubliez pas d'y adjoindre la procédure) :

```
'              ASC_TUCH.GFA
REPEAT
  mot_long%=BIOS(2,2)
  split_long(mot_long%)
  ascii|=octet_1|
  PRINT "Mot long    = ";mot_long%
  PRINT "Caractère   = ";CHR$(ascii|)
  PRINT "Code Ascii  = ";ascii|
  tch&=octet_3|
  PRINT "Code touche = ";tch&;" hexa = ";HEX$(tch&)
  PRINT "Code cplmt  = ";octet_4|
  analyse_cplmt(octet_4|)
  PRINT
UNTIL ascii|=13        ! Sortie : RETURN ou ENTER
'
PROCEDURE analyse_cplmt(octet|)
  IF octet|>7
    PRINT "Alternate"
    SUB octet|,8
  ENDIF
  IF octet|>3
    PRINT "Control"
    SUB octet|,4
  ENDIF
  IF octet|>1
    PRINT "Shift"
    SUB octet|,2
  ENDIF
RETURN
```

Il s'agit d'une boucle d'attente, qui se termine si l'utilisateur frappe la touche RETURN ou la touche ENTER (toutes deux génératrices du même code ASCII : 13). Lorsque vous pressez une touche, le programme vous indique quel est le mot long retourné par la fonction BIOS(2,2) ; le caractère (éventuel) ; le code ASCII (éventuel) ; le code de la touche (suivi de sa valeur en hexadécimal, notion abordée plus bas).

Pour voir les subtiles différences de codage, pressez d'abord la touche 'a' ; les touches SHIFT et 'a' ; CONTROL et 'a' ; ALTERNATE et 'a' ; CONTROL, SHIFT et 'a', etc.

Le mot long retourné par BIOS(2,2) est structuré comme ceci : l'octet 1 (à droite !) contient le code ASCII, l'octet 2 est nul, l'octet 3 contient le code touche, et l'octet 4 contient le code complément. Cette structure de mot long est identique à celle retournée par GEMDOS(1) ou



KEYGET, vue précédemment.
Codage du code complément : entrons dans les détails du code complément, qui vous permettra de savoir si l'utilisateur a pressé la touche SHIFT, la touche CONTROL, ou la touche ALTERNATE. Ce codage permet aussi de savoir s'il y a eu pression *simultanée* de deux de ces touches. Pourquoi pas les trois ? Parce que presser sur SHIFT, CONTROL et ALTERNATE... interrompt l'interpréteur !

Le bit 0 de l'octet code complément, correspond à la touche SHIFT ; le bit 1, à la touche CONTROL ; le bit 2, à la touche ALTERNATE. Lorsque l'un de ces bits est à '1', la touche correspondante a été pressée.

La procédure 'analyse_cplmt', qui appartient au programme précédent, illustre une gestion possible du code complément, et vous pouvez vous en inspirer (d'autres méthodes feraient appel à des tests de bit).

Comment savoir s'il y a eu SHIFT : nous venons de voir une façon de tester si la touche SHIFT a été pressée. Il existe un second moyen, beaucoup moins général.

Sur le schéma de codage des codes de touche (attention : il ne s'agit *pas* des codes complémentaires !), lorsqu'une touche comporte deux numéros, le numéro du dessus correspond au code touche obtenu lorsqu'une des touches SHIFT est simultanément enfoncée. Il est donc possible, dans certains cas, de savoir si SHIFT était enfoncée, en faisant un test portant uniquement sur le code touche.

*Décodage de INKEY$* : voici l'équivalent de ce que nous venons de voir appliqué à l'instruction INKEY$, dont les limites ont été précisées :

```
'              INKEY2.GFA
REPEAT
  deux_car$=INKEY$
  asc|=ASC(RIGHT$(deux_car$))
  scan|=ASC(LEFT$(deux_car$))
  PRINT deux_car$,LEN(deux_car$),"Ascii = ";asc|;" Scan =
";scan|
  PAUSE 10
UNTIL asc|=13
```

Cet exemple illustre le codage sur un mot (deux octets) qui caractérise cette fonction.

Conversion de l'hexadécimal

La notation hexadécimale - dont il a été brièvement question - n'est rien d'autre qu'un autre mode de numérotation et d'écriture. C'est une autre façon de dire les mêmes choses ! Vous la rencontrerez souvent dans les documentations de programmation et notamment pour donner la valeur des codes de touche, c'est pourquoi nous allons en effectuer un bref survol.

Les caractères employés, pour écrire une valeur hexadécimale, sont au nombre de 16 ; en voici la liste : "0123456789ABCDEF". La valeur "12A" est une valeur hexadécimale, tout comme "FFF" ou "10". Ce dernier exemple vous montre, que lorsqu'une valeur hexadécimale ne comporte que des chiffres, l'ambiguïté avec une valeur décimale est possible !
Nous n'entrerons pas dans les détails de cette notation, nous nous contenterons de vous rappeler, qu'il existe deux moyens à votre disposition pour manipuler facilement l'hexadécimal :

- la fonction HEX$: elle apparaît dans un de nos programmes et retourne une chaîne de caractères. Cette chaîne de caractères équivaut à la conversion d'une valeur décimale, qui est transmise à la fonction sous forme hexadécimale ;

- l'écriture de valeurs hexadécimales est conditionnée par l'emploi des caractères : '&H'. Quand ces caractères précédent une valeur, celle-ci est considérée comme codifiée sous la notation hexadécimale.

Voici un exemple vous montrant comment passer facilement d'une notation à l'autre :

```
PRINT "Décimal ";10              ! Décimal
PRINT "Décimal ";&H10            ! Hexadécimal
PRINT "Hexa (10) = ";HEX$(10)    ! Décimal (10) -> hexa
```

Rappel : la valeur '10' ne correspond pas à la même valeur dans la notation décimale (valeur : "dix"), et dans la notation hexadécimale (valeur : "seize"). Pour information, voici un schéma des codes des touches du clavier exprimés en

54 55 56 57 58 59 5A 5B 5C 5D
3B 3C 3D 3E 3F 40 41 42 43 44
78 79 7A 7B 7C 7D 7E 7F 80 81 82 83
01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 29 0E 62 61 63 64 65 66
0F 10 11 12 13 14 15 16 17 18 19 1A 1B 53 52 48 47 67 68 69 4A
1D 1E 1F 20 21 22 23 24 25 26 27 28 1C 2B 73 4B 50 74 4D 6A 6B 6C 4E
2A 60 2C 2D 2E 2F 30 31 32 33 34 35 36 6D 6E 6F 72
38 39 3A 70 71

CODES DES TOUCHES (hexadécimal)

hexadécimal : vous les trouverez souvent sous cette forme : Vous souhaitez en savoir plus sur la notation hexadécimale ? Reportez-vous au COLLECTOR consacré à l'assembleur, vous saurez tout !

Petits exemples

Ce premier exemple vous montre comment gérer des entrées clavier, qui sont censées déterminer, si l'utilisateur a frappé une touche de fonction (avec éventuelle pression simultanée sur la touche SHIFT), ou s'il a frappé une touche fléchée ; pour sortir il faut frapper RETURN ou ENTER (n'oubliez pas d'inclure la procédure 'split_long') :

```
'             FRAPPE.GFA
DO
  PRINT "Pressez une touche..."
  PRINT
  gere_clavier(BIOS(2,2))
LOOP
'
PROCEDURE gere_clavier(mot%)
  LOCAL tch|
  split_long(mot%)
  tch|=octet_3|
  IF (tch|>58 AND tch|<69) OR (tch|>83 AND tch|<94)
    PRINT "Touche de fonction"
  ENDIF
  IF tch|=72 OR tch|=75 OR tch|=77 OR tch|=80
    PRINT "Touche fléchée"
  ENDIF
  IF octet_1|=13
    END
  ENDIF
RETURN
```

Passons à notre second exemple : vous avez été familiarisé avec la notion d'entrée filtrée, dès notre première série d'articles (COLLECTOR 1 sur le GfA). Grâce aux nouvelles possibilités offertes par la gestion de clavier, telle que nous l'avons présentée ci-dessus, nous disposons d'autres techniques de filtrage. L'exemple qui va suivre demande à l'utilisateur de saisir quatre chiffres ; toute autre touche frappée engendre un message d'erreur, tant que quatre chiffres ne sont pas saisis (n'oubliez pas d'inclure la procédure 'split_long') :

```
'              FILTRE.GFA
filtrage
'
PROCEDURE filtrage
  LOCAL car%
  '
  PRINT "Entrez quatre chiffres SVP..."
  PRINT
  car%=0
  REPEAT
    split_long(BIOS(2,2))
    IF octet_1|>47 AND octet_1|<58  ! Chiffre ok
      INC car%
      PRINT CHR$(octet_1|);          ! Edition
    ELSE
      er_chiffre                     ! Caractère non
numérique
    ENDIF
  UNTIL car%=4
  PRINT
  PRINT "Merçi !"
RETURN
```

```
PROCEDURE er_chiffre
  LOCAL r%
  ALERT 1,"Que des chiffres SVP !",1,"Ok",r%
RETURN
```

Le test repose sur les codes ASCII des chiffres, compris entre 48 (chiffre 0) et 57 (chiffre 9). Attention : ne confondez pas les chiffres normalisés, avec les caractères, dont les codes ASCII sont compris entre 16 et 27 (caractères numériques non normalisés, de type "chiffres de calculatrice") ! Nous avons utilisé une saisie sans écho, pour des raisons qui doivent être évidentes.

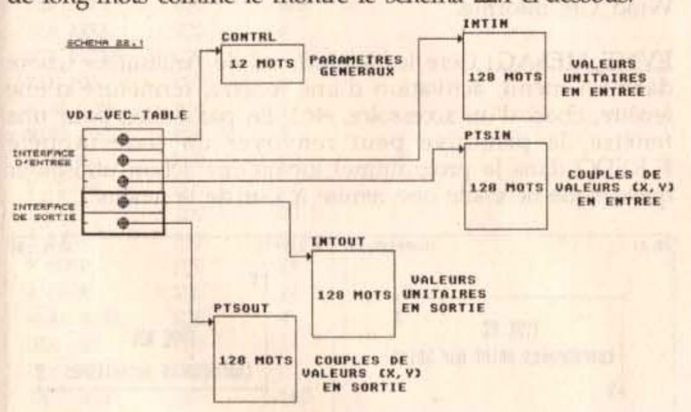
Suggestion : qu'il nous soit permis, sans vous commander, de revenir sur l'émulateur VT 52, dont il fut question à la fin de l'article 5 de cette série. Vous en savez assez pour effectuer une gestion des déplacements du curseur, avec entrées éventuelles d'informations !

Vous n'avez plus aucune excuse pour vous détourner de votre clavier. Certes, il apparaît comme moins spectaculaire que les menus, les messages d'alerte et autres gadgets informatiques - mais il est irremplaçable dans de nombreuses occasions. Par ailleurs, les instructions d'entrée "évoluées" comme INPUT, par exemple, ne permettent pas de tout faire ; il est notamment difficile de leur faire effectuer des entrées filtrées réellement sélectives. Dans ce cas, c'est à vous de "descendre" d'un niveau, pour construire vos propres routines de saisie, parfaitement adaptées à votre goût maniaque de perfection...

INITIATION A L'ASSEMBLEUR (XXII)

La fois dernière, nous avions présenté un premier exercice de manipulation de fenêtres sous GEM à l'aide des primitives de l'AES. Aujourd'hui, nous allons continuer la présentation de l'AES et aborder la deuxième composante du GEM : la VDI, "Virtual Device Interface", contenant les routines graphiques de base de la machine. Pour ce faire, nous allons compléter l'exercice du mois précédent, ce qui ne manquera pas de faire plaisir à ceux qui avaient tenté de modifier la taille de la fenêtre du programme ou encore de l'afficher "plein écran" et qui étaient tombés nez à nez avec une boîte d'alerte précisant qu'il fallait attendre le prochain numéro pour la suite... C'est chose faite !

Tout comme nous l'avions fait pour l'AES, nous allons tout d'abord décrire les phases de passage de paramètres jusqu'à arriver à l'appel effectif par la trappe logicielle n°2 (voir ST Mag 48). L'interface entre un programme et la VDI se présente sous forme d'un ensemble de tableaux de mots et de long-mots comme le montre le schéma 22.1 ci-dessous.



L'INTERFACE PROGRAMME / VDI

On distingue :

- contrl : tableau de 12 mots de contrôle général relatifs à la fonction et aux nombres de paramètres d'entrées-sorties dans les autres tableaux ;
- intin : tableau de 128 mots contenant les éventuels paramètres en entrée, le nombre de ces paramètres est précisé à la VDI par Contrl[3] ;
- intout : Même rôle que le tableau précédent mais pour les paramètres de sortie, le nombre des paramètres étant dans contrl[4] ;
- ptsin et ptsout : deux tableaux de 128 mots dans lesquels on retrouve respectivement des couples de valeurs (X,Y) en entrée et en sortie. En effet, la VDI étant chargée des gestions graphiques de bas niveau, beaucoup de primitives attendent et/ou retournent des couples de valeurs du type (abscisse, ordonnée).

Comme nous l'avons déjà signalé, la VDI est accessible par la trappe logicielle n°2. Son code (115 ou $0073) doit être passé au système dans D0.W et il convient de passer comme autre paramètre dans D1.L, l'adresse d'une table contenant les adresses des tableaux d'interface cités ci-dessus (voir schéma 22.1). Tout comme pour l'AES, où le code fonction était passé dans control[0], dans le cas de la VDI, ce code est dans Contrl[0]. Ainsi, d'un point de vue pratique, l'appel à la VDI se traduira par l'existence d'une section DATA et d'une section BSS comme suit :

```
               SECTION  DATA
vdi_vec_table  dc.l     contrl,
               dc.l     intin,ptsin
               dc.l     intout,ptsout

               SECTION  BSS
contrl         ds.w     12
intin          ds.w     128
intout         ds.w     128
ptsin          ds.l     128
ptsout         ds.l     128
```

Ces sections représentent respectivement le tableau d'adresses des variables (vdi_vec_table) et les variables mêmes. En ce qui concerne le sous-programme d'appel à la VDI (cf. le listing), celui que nous proposons charge simplement le registre D1 avec l'adresse de la table des tableaux de paramètres et évidemment D0 avec le code de la VDI. Le programme proposé consiste en la manipulation d'une fenêtre GEM classique : déplacement, affichage plein écran, et modification de la taille de la fenêtre. Ces manipulations simples sont à la base de tout développement GEM. En ce qui concerne l'intérieur de la fenêtre, les appels à la VDI assurent la remise à jour du contenu de la fenêtre. Le programme proposé n'effectue rien de particulier comme "traitement", en effet la routine do_in_fenetre ne

fait que "RTS". Loisir à vous de dessiner ou de gérer du texte à l'intérieur de la fenêtre... Mais oui, bien sûr, nous ferons cela ensemble une prochaine fois ! En attendant donc le nirvaña, le pseudo-code du programme proposé ce mois-ci est le suivant :

```
/* exécution du STARTUP */
/* installation de l'environnement AES */
/* récupération du numéro d'identification de l'application */
/* ouverture d'une station de travail VDI */
/* fixer la forme de la souris */
/* récupération des dimensions utiles de l'écran */
/* calcul des dimensions utiles de la fenêtre */
/* sauvegarde des coordonnées et dimensions de la fenêtre */
/* création de la fenêtre */
/* fixer le titre de la fenêtre */
/* calcul de la taille initiale d'affichage */
/* affichage de la fenêtre */
/* sauvegarde des coordonnées et dimensions de la fenêtre */
autre :
REPETER
  /* attente d'un événement en provenance de l'AES */
  /* récupération du message de l'AES */
  DECIDER SELON "le message"
   SI "demande de retraçage" ALORS /* va à retrace */
   SI "déplacement de la fenêtre" ALORS /* va à deplace */
   SI "affichage plein écran" ALORS /* va à plein_ecran */
   SI "redimensionnement fen." ALORS /* va à chg_taille */
   SI "fermeture de la fenêtre" ALORS /* va à fin */
  FIN DECIDER
JUSQU'A "infini"
retrace :
/* validation d'un rectangle de clipping */
/* choix du style de remplissage */
/* masquage de la souris */
/* traçage du rectangle intérieur de fenêtre */
/* gestion de l'intérieur de la fenêtre... */
/* affichage de la souris */
/* aller à autre */
deplace :
/* réaffichage de la fenêtre déplacée */
/* sauvegarde coordonnées et dimensions de la fenêtre */
/* aller à autre */
plein_ecran :
SI "la fenêtre n'est pas plein_ecran"
 ALORS
   /* sauvegarde coordonnées et dimensions dans [save] */
   /* récup. de l'espace écran pour affichage maximal */
   /* positionner [flag] plein_ecran */
 SINON
   /* remettre à zéro [flag] plein_ecran */
   /* récupérer coordonnées et dimensions de [save] */
 FINSI
/* sauvegarde des coordonnées et dimensions de la fenêtre */
/* affichage de la fenêtre (ancienne ou taille max.) */
/* aller à autre */
chg_taille :
/* récupération de la hauteur de la fenêtre */
SI "inférieur au minimum" ALORS /* fixer haut. à mini.*/
/* récupération de la largeur de la fenêtre */
SI "inférieur au minimum" ALORS /* fixer larg. à mini.*/
/* réafficher la nouvelle fenêtre */
/* sauvegarde coordonnées et dimensions de la fenêtre */
/* aller à autre */
fin :
/* fermeture de la fenêtre */
/* destruction de la fenêtre */
/* fermeture de la station de travail */
/* désinstallation de l'environnement AES */
/* fin */
```

L'ensemble des appels VDI et AES mis en oeuvre pour implémenter ce pseudo-code est répertorié sur le schéma 22.2 ci-dessous.

APPL_INIT INITIALISE LES VARIABLES GLOBALES GEM
APPL_EXIT LIBERE L'ESPACE MEMOIRE OCCUPE PAR L'AES
GRAF_HANDLE FOURNIT L'IDENTIFICATEUR DE LA STATION DE TRAVAIL OUVERTE PAR L'AES
GRAF_MOUSE FIXE LA FORME DU CURSEUR DE SOURIS
EVNT_MESAG ATTENTE D'UNE ACTION DE L'UTILISATEUR
WIND_CREATE ALLOCATION D'UNE FENETRE AVEC LES ATTRIBUTS PRECISES
WIND_OPEN OUVERTURE D'UNE FENETRE DEJA ALLOUEE
WIND_CALC CALCUL DES DIMENSIONS D'UNE FENETRE (AVEC OU SANS ATTRIBUTS)
WIND_GET LECTURE DES PARAMETRES ATTACHES A UNE FENETRE
WIND_SET MODIFICATIONS DE LA POSITION ET/OU DES ATTRIBUTS D'UNE FENETRE
WIND_CLOSE FERMETURE D'UNE FENETRE
WIND_DELETE DESALLOCATION D'UNE FENETRE
V_OPNVWK OUVERTURE D'UNE STATION DE TRAVAIL VDI
VS_CLIP VALIDATION/DEVALIDATION D'UNE ZONE DE "CLIPPING"
VSF_INTERIOR FIXE LE TYPE DE MOTIF DE REMPLISSAGE UTILISE
V_RECFL TRACAGE D'UN RECTANGLE PLEIN D'APRES LE MOTIF VALIDE
V_SHOW_C AFFICHAGE DU CURSEUR DE LA SOURIS
V_HIDE_C MASQUAGE DU CURSEUR DE LA SOURIS
V_CLSVWK FERMETURE D'UNE STATION DE TRAVAIL VDI

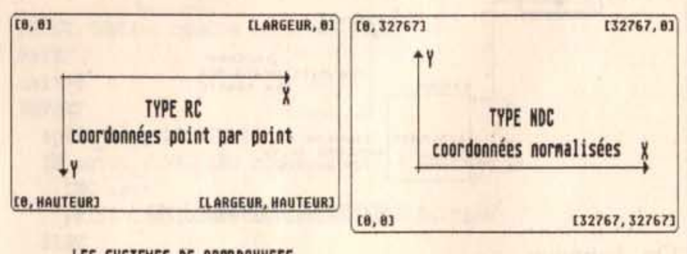
LES APPELS AES ET VDI POUR LE PROGRAMME

Certaines de ces primitives nécessitent un complément d'informations que nous vous proposons sans plus tarder...

WIND_GET: Permet de récupérer des informations essentielles sur une fenêtre (taille interne, externe, position des ascenceurs, ...) et par extension sur l'écran (principalement les dimensions utiles de la zone de travail, hormis le menu). Le paramètre dans int_in[1] permet de préciser selon sa valeur (douze possibilités) la fonctionnalité souhaitée.

WIND_SET: Offre pratiquement les mêmes possibilités que la primitive précédente mais opère des modifications sur la fenêtre (nouvelle taille, déplacement, ...) alors que Wind_Get informe.

EVNT_MESAG: Gère les interactions de l'utilisateur (choix dans un menu, activation d'une fenêtre, fermeture d'une fenêtre, choix d'un accessoire, etc.). En particulier, pour une fenêtre, la primitive peut renvoyer un code (appelé F_REDO dans le programme) lorsqu'une action utilisateur quelconque nécessite une remise à jour de la fenêtre.



LES SYSTEMES DE COORDONNEES

En ce qui concerne la VDI, il convient de noter les remarques suivantes :

V_OPNVWK: Ouvre une station de travail VDI, indispensable pour travailler avec la VDI. Onze

paramètres sont nécessaires, néanmoins, la plupart d'entre-eux sont fixés à 1. Ils représentent les paramètres de fonctionnement essentiels tels que le type de fontes de caractères, les couleurs (du texte, du fond, etc.). Le dernier paramètre, un peu particulier, représente le type de coordonnées utilisé : NDC (normalized Device Coordinates) ou RC (Raster coordinates). Le schéma 22.3 présente les différences entre ces deux types de représentation.

Le premier type présente l'avantage de la portabilité puisqu'il est indépendant du type de périphérique concerné (table traçante, écran, etc.) cependant il oblige à une conversion systématique puisque tout appel VDI attend des coordonnées de type RC...

VS_CLIP: Cet appel permet de valider ou de dévalider un rectangle de clipping, c'est-à-dire une zone en dehors de laquelle la VDI refusera d'intervenir... même si vous vous plantez dans le calcul des coordonnées ! C'est plus que pratique et cela évite la lourdeur des tests de bornes dans le programme d'application.

Enfin, il est important de ne pas oublier de masquer la souris avant toute opération de retraçage avec la VDI et bien sûr de la réafficher ensuite...

Les appels AES et VDI détaillés ci-dessus sont parmi les plus importants. Vous pourrez les récupérer pour le développement d'une application GEM conséquente. Le listing complet du programme proposé (ci-dessous) comporte les parties principales suivantes :

- Les équivalences
- Le code de démarrage (Startup)
- Le prg. principal (Main) et ses points de branchements
- Le sous-programme d'appel de l'AES
- Les appels AES utilisés pour le programme
- Le sous-programme d'appel de la VDI
- Les appels VDI utilisés pour le programme
- La zone de données initialisée (DATA)
- La zone de données non initialisée (BSS)

```
************* Diverses équivalences *******************
FORM_SOURIS     EQU     0
FEN_ATTR        EQU     $2F
FEN_AREA        EQU     4
FIX_TITLE       EQU     2
FIX_POS         EQU     5
WIND_COUR       EQU     5
WIND_FULL       EQU     7
F_REDO          EQU     20
F_CLOSE         EQU     22
F_ALL           EQU     23
F_SIZE          EQU     27
F_MOVE          EQU     28
F_DRAW          EQU     20
WORK_AREA       EQU     4
DIM_EXT         EQU     1
DIM_INT         EQU     0
HAUT_MIN        EQU     $60
LARG_MIN        EQU     $138

************* Codes de fonctions AES ******************
APPL_INIT       EQU     $0A
APPL_EXIT       EQU     $13
FORM_ALERT      EQU     $34
GRAF_HANDLE     EQU     77
EVNT_MESAG      EQU     23
WIND_GET        EQU     104
WIND_SET        EQU     105
WIND_CALC       EQU     108
WIND_CREATE     EQU     100
WIND_OPEN       EQU     101
WIND_CLOSE      EQU     102
WIND_DELETE     EQU     103
GRAF_MOUSE      EQU     78

************* Codes de fonctions VDI ******************
V_OPNVWK        EQU     100
V_CLSVWK        EQU     101
VS_CLIP         EQU     129
V_RECFL         EQU     114
VSF_INTERIOR    EQU     23
V_HIDE_C        EQU     123
V_SHOW_C        EQU     122

************* Codes de fonctions DOS ******************
SHRINK          EQU     $4A
TERM            EQU     $4C

*******************************************************
                SECTION TEXT
                move.l  4(a7),a0
                move.l  $C(a0),d0
                add.l   $14(a0),d0
                add.l   $1C(a0),d0
                add.l   #$800,d0
                andi.l  #-2,d0
                lea.l   0(a0,d0),a7
                move.l  d0,-(a7)
                move.l  a0,-(a7)
                clr.w   -(a7)
                move.w  #SHRINK,-(a7)
                trap    #1
                add.l   #12,a7
                tst.l   d0
                bne     Out_gem
                bsr     Main
Out_gem         move.w  d0,-(a7)
                move.w  #TERM,-(a7)
                trap    #1

*******************************************************
Main            equ     *
                bsr     Appl_Init
                move.w  d0,ap_id       identification GEM

* Récupération du "handle" pour les appels à la VDI
                bsr     Graf_Handle
                move.w  d0,this_hndl

* Ouverture d'une station de travail VDI
                bsr     V_OpnVWk

* Fixer la forme de la souris (flèche)
                move.w  #FORM_SOURIS,int_in
                bsr     Graf_Mouse

* Récupération de l'espace écran disponible
* (dans le tableau int_out[1..4])
                move.w  #0,int_in         0 = écran
                move.w  #WORK_AREA,int_in+2
                bsr     Wind_Get

* Calcul de la taille de la fenêtre à partir de ces paramètres
                move.w  #FEN_ATTR,int_in
                move.w  #DIM_EXT,int_in+2
                move.w  int_out+2,int_in+4
                move.w  int_out+4,int_in+6
                move.w  int_out+6,int_in+8
                move.w  int_out+8,int_in+10
                bsr     Wind_Calc
```

```
* Sauvegarde des dimensions de la fenêtre
        move.w   int_out+2,org_x
        move.w   int_out+4,org_y
        move.w   int_out+6,long_x
        move.w   int_out+8,haut_y

* Création de la fenêtre
        move.w   #FEN_ATTR,int_in
        move.w   int_out+2,int_in+2
        move.w   int_out+4,int_in+4
        move.w   int_out+6,int_in+6
        move.w   int_out+8,int_in+8
        bsr      Wind_Create
        move.w   d0,my_handle

* Inscription du titre de la fenêtre
        move.w   my_handle,int_in
        move.w   #FIX_TITLE,int_in+2
        move.l   #titre_fen,int_in+4
        bsr      Wind_Set

* Affichage de la fenêtre, Calcul d'une taille initiale
        move.w   my_handle,int_in
        move.w   org_x,int_in+2
        move.w   org_y,int_in+4
        move.w   long_x,int_in+6
        move.w   haut_y,int_in+8
        addi.w   #40,int_in+2
        addi.w   #40,int_in+4
        subi.w   #70,int_in+6
        subi.w   #70,int_in+8
        bsr      Wind_Open

* Récupération des dimensions de la fenêtre
sauv_taille  move.w   my_handle,int_in
        move.w   #FEN_AREA,int_in+2
        bsr      Wind_Get

* Sauvegarde des dimensions de la fenêtre
        move.w   int_out+2,org_x
        move.w   int_out+4,org_y
        move.w   int_out+6,long_x
        move.w   int_out+8,haut_y

* Boucle principale d'attente,des messages en provenance de
l'AES
autre_mess   move.l   #buf_mess,addr_in
        bsr      Evnt_Mesag
        lea      buf_mess,a0
        move.w   (a0),d0         Type du message reçu
        cmpi.w   #F_REDO,d0
        beq      retrace         si demandé par l'AES
        cmpi.w   #F_MOVE,d0
        beq      deplace         si déplacement
        cmpi.w   #F_ALL,d0
        beq      plein_ecran     si plein écran
        cmpi.w   #F_SIZE,d0
        beq      chg_taille      si chgmt taille
        cmpi.w   #F_CLOSE,d0
        beq      fin             si fermeture
        bra      autre_mess

* Fermeture et destruction de la fenêtre
fin     move.w   my_handle,int_in
        bsr      Wind_Close
        move.w   my_handle,int_in
        bsr      Wind_Delete

* Fermeture de la station de travail VDI
* Désinstallation de l'environnement AES
        bsr      V_ClsVWk
        rts

************* Fin du Main *****************************
deplace      move.w   6(a0),d0
        cmp.w    my_handle,d0
        bne      autre_mess
        move.w   8(a0),int_in+4   nouvelle pos. X
        move.w   10(a0),int_in+6  nouvelle pos. Y
        move.w   12(a0),int_in+8  nouvelle larg.
        move.w   14(a0),int_in+10 nouvelle haut.

* Affichage de fenêtre à sa nouvelle position
trace        move.w   my_handle,int_in
        move.w   #FIX_POS,int_in+2
        bsr      Wind_Set
        bra      sauv_taille
*------------------------------------*
plein_ecran  tst.b    full_flag    Déjà Plein écran ?
        bne      old_fen
        move.w   my_handle,int_in
        move.w   #WIND_COUR,int_in+2
        bsr      Wind_Get

* sauvegarde des dimensions de la fenêtre courante
        move.w   int_out+2,save_x
        move.w   int_out+4,save_y
        move.w   int_out+6,save_w
        move.w   int_out+8,save_h

* récupération de la taille maximale de la fenêtre
        move.w   #0,int_in
        move.w   #WIND_FULL,int_in+2
        bsr      Wind_Get
        move.w   int_out+2,int_in+4
        move.w   int_out+4,int_in+6
        move.w   int_out+6,int_in+8
        move.w   int_out+8,int_in+10
        st       full_flag
        bra      all_fen
old_fen      sf       full_flag
        move.w   save_x,int_in+4   nouvelle pos. X
        move.w   save_y,int_in+6   nouvelle pos. Y
        move.w   save_w,int_in+8   nouvelle larg.
        move.w   save_h,int_in+10  nouvelle haut.
all_fen      bra      trace
*------------------------------*
chg_taille   move.w   6(a0),d0
        cmp.w    my_handle,d0
        bne      autre_mess
        move.w   8(a0),int_in+4    nouvelle pos. X
        move.w   10(a0),int_in+6   nouvelle pos. Y

* tests sur dimensions min. en hauteur et en largeur
        move.w   12(a0),d0
        cmpi.w   #LARG_MIN,d0
        bhi      .larg_ok
        move.w   #LARG_MIN,d0
.larg_ok     move.w   d0,int_in+8      nouvelle largeur
        move.w   14(a0),d0
        cmpi.w   #HAUT_MIN,d0
        bhi      .haut_ok
        move.w   #HAUT_MIN,d0
.haut_ok     move.w   d0,int_in+10     nouvelle hauteur
        bra      trace
retrace      movem.w  8(a0),d0-d3      bloc de coordonnées

* traitement du message de remise à jour envoyé
* par l'AES à l'application.. (event_messag)
        add.w    d0,d2  d0="x depart" d2="x fin"
        add.w    d1,d3  d1="y depart" d3="y fin"

* Validation d'un rectangle de clipping
        bsr      Vs_Clip
```

```
* Style de remplissage de l'intérieur du rectangle
        bsr     Vsf_Interior

* Masquage de la souris (avant tout traçage effectif)
        bsr     V_Hide_C

* Retraçage du rectangle à l'intérieur de la fenêtre
        movem.w org_x,d0-d3 Bloc de coordonnées
        add.w   d0,d2       d0="x depart" d2="x fin"
        subq.w  #1,d2  ...

* Pour laisser apparent le bord intérieur horizontal de la
fenêtre
        add.w   d1,d3  d1="y depart" d3="y fin"
        subq.w  #1,d3  ...

* Pour laisser apparent le bord intérieur vertical de la
fenêtre
        bsr     V_RecFl

* Gestion de l'intérieur de la fenêtre, A vous de jouer...
        bsr     do_in_fenetre

* Affichage de la souris (Après le traçage)
        bsr     V_Show_C
        bra     autre_mess
do_in_fenetre nop   Tout reste
        nop     à faire...
        rts

************************************************
* Appel AES                                    *
************************************************
AES     equ     *
        lea     control,a1
        move.w  d0,(a1)     numéro de fonction
        clr.w   8(a1)
        move.l  #aes_vec_table,d1
        move.w  #200,d0
        trap    #2
        move.w  int_out,d0
        rts
* * * * * * * * * * * * * * * * * * * * * * * * * *
Appl_Init      equ     *
        lea     control,a1
        clr.w   2(a1)
        move.w  #1,4(a1)
        clr.w   6(a1)
        moveq   #APPL_INIT,d0
        bsr     AES
        rts
* * * * * * * * * * * * * * * * * * * * * * * * * *
Appl_Exit      equ     *
        lea     control,a1
        clr.w   2(a1)
        move.w  #1,4(a1)
        clr.w   6(a1)
        moveq   #APPL_EXIT,d0
        bsr     AES
        rts
* * * * * * * * * * * * * * * * * * * * * * * * * *
Evnt_Mesag     equ     *
        lea     control,a1
        clr.w   2(a1)
        move.w  #1,4(a1)
        move.w  #1,6(a1)
        moveq   #EVNT_MESAG,d0
        bsr     AES
        rts
* * * * * * * * * * * * * * * * * * * * * * * * * *
Graf_Handle    equ     *
        lea     control,a1
        clr.w   2(a1)
        move.w  #5,4(a1)
        clr.w   6(a1)
        moveq   #GRAF_HANDLE,d0
        bsr     AES
        rts
* * * * * * * * * * * * * * * * * * * * * * * * * *
Graf_Mouse     equ     *
        lea     control,a1
        move.w  #1,2(a1)
        move.w  #1,4(a1)
        move.w  #1,6(a1)
        moveq   #GRAF_MOUSE,d0
        bsr     AES
        rts
* * * * * * * * * * * * * * * * * * * * * * * * * *
Wind_Get       equ     *
        lea     control,a1
        move.w  #2,2(a1)
        move.w  #5,4(a1)
        clr.w   6(a1)
        moveq   #WIND_GET,d0
        bsr AES
        rts
* * * * * * * * * * * * * * * * * * * * * * * * * *
Wind_Calc      equ     *
        lea     control,a1
        move.w  #6,2(a1)
        move.w  #5,4(a1)
        clr.w   6(a1)
        moveq   #WIND_CALC,d0
        bsr     AES
        rts
* * * * * * * * * * * * * * * * * * * * * * * * * *
Wind_Create    equ     *
        lea     control,a1
        move.w  #5,2(a1)
        move.w  #1,4(a1)
        clr.w   6(a1)
        moveq   #WIND_CREATE,d0
        bsr     AES
        rts
* * * * * * * * * * * * * * * * * * * * * * * * * *
Wind_Set       equ     *
        lea     control,a1
        move.w  #6,2(a1)
        move.w  #1,4(a1)
        clr.w   6(a1)
        moveq   #WIND_SET,d0
        bsr     AES
        rts
* * * * * * * * * * * * * * * * * * * * * * * * * *
Wind_Open      equ     *
        lea     control,a1
        move.w  #5,2(a1)
        move.w  #1,4(a1)
        clr.w   6(a1)
        moveq   #WIND_OPEN,d0
        bsr     AES
        rts
* * * * * * * * * * * * * * * * * * * * * * * * * *
Wind_Close     equ     *
        lea     control,a1
        move.w  #1,2(a1)
        move.w  #1,4(a1)
        clr.w   6(a1)
        moveq   #WIND_CLOSE,d0
        bsr     AES
        rts
* * * * * * * * * * * * * * * * * * * * * * * * * *
Wind_Delete    equ     *
        lea     control,a1
```

```
        move.w  #1,2(a1)
        move.w  #1,4(a1)
        clr.w   6(a1)
        moveq   #WIND_DELETE,d0
        bsr     AES
        rts

*******************************************************
* Appels VDI                                          *
*******************************************************
VDI     equ     *
        move.l  #vdi_vec_table,d1
        move.w  #115,d0
        trap    #2
        rts
V_OpnVWk equ    *
        move.w  #V_OPNVWK,contrl   code fct VDI
        move.w  #0,contrl+2  tableau ptsin vide
        move.w  #11,contrl+6      11 val. intin
        move.w  this_hndl,contrl+12

* Initialisation des onze valeurs du tableau intin
* les dix premières valeurs sont égales à 1, la onzième valeur
* indique le choix des coordonnées de type RC
* Rem : cet appel fixe la couleur par défaut à "blanc"

        lea     intin,a0
        moveq   #9,d0   dix boucles à effectuer
.bcl    move.w  #1,(a0)+
        dbf     d0,.bcl
        move.w  #2,(a0)           coordonnées RC
        bsr     VDI
        rts
* * * * * * * * * * * * * * * * * * * * * * * * * * * *
V_ClsVWk equ    *
        move.w  #V_CLSVWK,contrl         code VDI
        move.w  #0,contrl+2  tableau ptsin vide
        move.w  #0,contrl+6  tableau intin vide
        move.w  this_hndl,contrl+12
        bsr     VDI
        rts
* * * * * * * * * * * * * * * * * * * * * * * * * * * *
Vs_Clip equ     *
        move.w  #VS_CLIP,contrl          code VDI
        move.w  #2,contrl+2  2 couples -> ptsin
        move.w  #1,contrl+6     1 val. -> intin
        move.w  this_hndl,contrl+12

* init. des valeurs de intin et ptsin
        move.w  #1,intin    activer le clipping
        move.w  d0,ptsin      Coin sup. gauche X
        move.w  d1,ptsin+2  Coin sup. gauche Y
        move.w  d2,ptsin+4    Coin inf. droit X
        move.w  d3,ptsin+6    Coin inf. droit Y
        bsr     VDI
        rts
* * * * * * * * * * * * * * * * * * * * * * * * * * * *
Vsf_Interior equ *
        move.w  #VSF_INTERIOR,contrl  code VDI
        move.w  #0,contrl+2       tab. ptsin vide
        move.w  #1,contrl+6     1 val. -> intin
        move.w  this_hndl,contrl+12

* initialisation des valeurs intin et ptsin
        move.w  #1,intin     couleur par défaut
        bsr     VDI
        rts
* * * * * * * * * * * * * * * * * * * * * * * * * * * *
V_Show_C equ    *
        move.w  #V_SHOW_C,contrl         code VDI
        move.w  #0,contrl+2       tab. ptsin vide
        move.w  #1,contrl+6     1 val. -> intin
        move.w  this_hndl,contrl+12
        move.w  #0,intin    Aff. inconditionnel
        bsr     VDI
        rts
* * * * * * * * * * * * * * * * * * * * * * * * * * * *
V_Hide_C equ    *
        move.w  #V_HIDE_C,contrl         code VDI
        move.w  #0,contrl+2     tab. ptsin vide
        move.w  #0,contrl+6     tab. intin vide
        move.w  this_hndl,contrl+12
        bsr     VDI
        rts
* * * * * * * * * * * * * * * * * * * * * * * * * * * *
V_RecFl equ     *
        move.w  #V_RECFL,contrl          code VDI
        move.w  #2,contrl+2  2 couples -> ptsin
        move.w  #0,contrl+6     tab. intin vide
        move.w  this_hndl,contrl+12
* initialisation des valeurs intin et ptsin
        move.w  d0,ptsin      Coin sup. gauche X
        move.w  d1,ptsin+2  Coin sup. gauche Y
        move.w  d2,ptsin+4    Coin inf. droit X
        move.w  d3,ptsin+6    Coin inf. droit Y
        bsr     VDI
        rts

*******************************************************
        SECTION DATA
aes_vec_table dc.l  control,global
        dc.l    int_in,int_out
        dc.l    addr_in,addr_out
vdi_vec_table dc.l  contrl
        dc.l    intin,ptsin
        dc.l    intout,ptsout

titre_fen dc.b  ' Porte ouverte'
        dc.b    ' sur les fenêtres... ',0

*******************************************************
        SECTION BSS

* Pour l'AES
control     ds.w    5
global      ds.w    14
int_in      ds.w    16
int_out     ds.w    7
addr_in     ds.l    3
addr_out    ds.l    1
* Pour la VDI
contrl      ds.w    12
intin       ds.w    128
intout      ds.w    128
ptsin       ds.w    128
ptsout      ds.w    128

* Pour l'application
org_x       ds.w    1
org_y       ds.w    1
long_x      ds.w    1
haut_y      ds.w    1
save_x      ds.w    1
save_y      ds.w    1
save_w      ds.w    1
save_h      ds.w    1
this_hndl   ds.w    1
my_handle   ds.w    1
ws_handle   ds.w    1
ap_id       ds.w    1
buf_mess    ds.b    16
full_flag   ds.b    1    (...)

            END
```



PARIS 10e 42.06.50.50
LYON 1er 72.00.96.96

LE GRAND SPECIALISTE ATARI

ATARI PRO

GRAPHISME

Spectrum 512	595 FTTC
Sprite animator	390 FTTC
Creator dessin	990 FTTC
Arabesque	990 FTTC
Cybercontrol	595 FTTC
Stad	800 FTTC
Cyberpaint 2.0	695 FTTC
Cybersculpt	895 FTTC
Cyberstudio	800 FTTC
Unispec	595 FTTC
Cybertexture	595 FTTC
ZZ-Lazypaint	595 FTTC
Didot	690 FTTC
Image	490 FTTC
Dessin technique	1490 FTTC

BASE DE DONNEES

Super base II	890 FTTC
Super base pro	1490 FTTC
Super base pro III	2490 FTTC
Graal base	1590 FTTC

TRAITEMENT DE TEXTES

Le rédacteur 3	990 FTTC
Beckertext	750 FTTC
Script	790 FTTC
Calligrapheur pro	1490 FTTC
Signum 1.2.0	1490 FTTC
Fontz 1	299 FTTC
1st Word +	990 FTTC

TABLEUR

Backer calc	990 FTTC
Graal calc	790 FTTC
Kspread 4	990 FTTC

MISE EN PAGE

Calamus "The Must"	2490 FTTC
Publishing partner	2490 FTTC
Timeworks publisher	990 FTTC

COMPTABILITE

Le Comptable II	790 FTTC
Gestocks 90	1779 FTTC
Le Gestionnaire	590 FTTC
Le Dentiste	5990 FTTC
Le Professeur	790 FTTC

LANGAGE

Oss Pascal SI	800 FTTC
Personal interpreteur	
Nouvel interpreteur	595 FTTC
CV 2.0 "professionnel"	
GFA assembleur	590 FTTC
A-Debog	590 FTTC
Hisoft Wercs	360 FTTC
Hisoft devpac St	750 FTTC

DIVERS

Flexidump plus	490 FTTC
Starter pack	590 FTTC
Pack bureautique	990 FTTC
PC globes 3.0	490 FTTC

MUSIQUE

ST Replay 8	790 FTTC
Track 24	499 FTTC
Big boss +	890 FTTC
Pro 12	690 FTTC
Pro 24	2600 FTTC
Supase ver. 2.0	3990 FTTC

LOGICIELS ATARI

EDUC NATHAN

EVEIL	
Apprends moi à lire maternelle/CP	295 FTTC
Apprends moi à compter maternelle/CP	275 FTTC
Apprends moi à lire 2 CP	295 FTTC
Apprends moi à écrire 1 et 2 CP/CE	275 FTTC
Donald et l'Alphabet magique 3/6 ans	275 FTTC
Le Train Express de Dingo - 3/6 ans	275 FTTC
Mickey et le Zoo en folie - 3/6 ans	275 FTTC
La machine à Mots-Croisés de Mickey D - 6/8 ans	275 FTTC
Les 3 Petits cochons - 4/6 ans	220 FTTC
Les Castors Juniors - 9/11 ans	220 FTTC
Picsou chasseur de Trésor 9/12 ans	220 FTTC
FRANÇAIS	
Nathan Ecole Français - CP/CE1	275 FTTC
Nathan Ecole Français - CE1/CE2	275 FTTC
Nathan Ecole Français - CM1	275 FTTC
Nathan Ecole Français - CM2	275 FTTC
Langue française CE1	275 FTTC
Langue française CE2	275 FTTC
Langue française CM1	275 FTTC
Langue française CM2	275 FTTC
ExoNathan Français - 6e	275 FTTC
ExoNathan Français - 5e	275 FTTC
ExoNathan Français - 4e	275 FTTC
ExoNathan Français - 3e	275 FTTC
Langue française 6e	275 FTTC
Langue française 5e	275 FTTC
Langue française 4e	275 FTTC
Langue française 3e	275 FTTC
Français - Résumés - 1ère	275 FTTC
Français - Résumés - 2nde	275 FTTC
Français - Commentaires - 1ère	275 FTTC
Français - Commentaires - 2nde	275 FTTC
Ecrire sans fautes Vol. 1 6e/3e	275 FTTC
Les sons difficiles	
Ecrire sans fautes Vol. 2 6e/3e	275 FTTC
Les noms, adjectifs, verbes	
Labyrinthe d'Orthophus	275 FTTC
Labyrinthe Lexicos	275 FTTC
JOKER - Expression écrite 6e	295 FTTC
JOKER - Orthographe/Grammaire 6e	295 FTTC
MATHEMATIQUES	
Nathan Ecole Maths - CP/CE1	275 FTTC
Nathan Ecole Maths - CE2	275 FTTC
Nathan Ecole Maths - CM1	275 FTTC
Nathan Ecole Maths - CM2	275 FTTC
Labyrinthe d'Errore - 11/14 ans	275 FTTC
Maths - 6e/5e	275 FTTC
ExoNathan Maths - 6e	275 FTTC
ExoNathan Maths - 5e	275 FTTC
ExoNathan Maths - 4e	275 FTTC
ExoNathan Maths - 3e	275 FTTC
Mathématiques - 4e	275 FTTC
Mathématiques - 3e	275 FTTC
Labyrinthe aux Cents Calculs	275 FTTC
Labyrinthe aux 1000 Calculs 11/14 ans	275 FTTC
Géométrie Débutant	275 FTTC
Géométrie Confirmée	275 FTTC
JOKER - Maths 6e	295 FTTC
ANGLAIS	
Anglais débutant	275 FTTC
Anglais confirmé	275 FTTC
JOKER - Anglais 6e	295 FTTC
Anglais perfectionnement 2nde	275 FTTC
Anglais perfectionnement 1ère	275 FTTC
Labyrinthe Anglomania (11/14 ans) Collège	275 FTTC
Labyrinthe Anglomania (8/10 ans) Ecole	275 FTTC
MICRO BREVET	
MicroBrevet Français	220 FTTC
MicroBrevet Histoire	220 FTTC
MicroBrevet Géographie	220 FTTC
MICRO BAC	
MicroBac Français 1ère/term	220 FTTC
MicroBac Anglais 1ère/term	220 FTTC
MicroBac Espagnol 1ère/term	220 FTTC
MicroBac Allemand 1ère/term	220 FTTC
MicroBac Géographie 1ère/term	220 FTTC
MicroBac Histoire 1ère/term	220 FTTC
ALLEMAND	
Allemand débutant 1ère année	275 FTTC
Allemand confirmé	275 FTTC
Allemand perfectionnement 2nde	275 FTTC
Allemand perfectionnement 1ère	275 FTTC
HISTOIRE	
Labyrinthe des Pharaons	275 FTTC

EDUC ADI

CM1 - Français + environnement	265 FTTC
CM1 - Maths + environnement	265 FTTC
CM2 - Français + environnement	265 FTTC
CM2 - Maths + environnement	265 FTTC
6e - Français + environnement	265 FTTC
6e - Maths + environnement	265 FTTC
5e - Français + environnement	265 FTTC
5e - Maths + environnement	265 FTTC
5e - Anglais + environnement	265 FTTC
ADI Français	195 FTTC
ADI Maths	195 FTTC
ADI Anglais	195 FTTC

EDUC VTA

Orthogus Tome 1 (CM-Collège) (grammaire orthographe)	245 FTTC
Orthogus Tome 2 (Collège)	245 FTTC
Mathex CE/CM	245 FTTC
Mathex 6e/5e	245 FTTC
Mathex 4e/3e	245 FTTC
Mathex 6e/3e	295 FTTC
Révision Orthographe/Grammaire/ Mathex 6e/5e	350 FTTC
Mathex 4e/3e	350 FTTC

COMPILATIONS

Cartoons	275 FTTC
Chevalier	275 FTTC
Sega arcade turbo	245 FTTC
Aventuriers	245 FTTC
Star d'hollywood	245 FTTC
Weels of Fire	295 FTTC
Justicier 2	269 FTTC
Challenger	295 FTTC
Epyx Sporting gold	245 FTTC
Monde des merveilles	259 FTTC
Guerriers Ninja	290 FTTC
Justiciers	249 FTTC
Edition n° 1	249 FTTC

JEUX

Power monger	245 FTTC
Gazza 2	235 FTTC
Dragon's air 2	395 FTTC
Pick'n pile	245 FTTC
World Champion Ship Soccer	225 FTTC
Battle Command	195 FTTC
ADS	275 FTTC
Crime wave	245 FTTC
Dick Tracy	245 FTTC
E swat	245 FTTC
Lemmings	245 FTTC
Line of fine	245 FTTC
Murder	245 FTTC
Narc	195 FTTC
Pang	195 FTTC
Puzznic	220 FTTC
Robocop 2	245 FTTC
Chase HQ 2	195 FTTC
Sliders	245 FTTC
Team Suzuki	245 FTTC
Toki	245 FTTC
Total Recall	245 FTTC
Art de la guerre	295 FTTC
Battle Storm	265 FTTC
Carthage	245 FTTC
Cougar Force	265 FTTC
Disc	245 FTTC
Intuder	295 FTTC
Gheisha	260 FTTC
Bande à Picsou	245 FTTC
Crime ne paie pas	295 FTTC
Monkey islands	245 FTTC
MUDS	245 FTTC
Murders in space	225 FTTC
Mystical	235 FTTC
Night breed	235 FTTC
Night shift	245 FTTC
Tortues Ninja	245 FTTC
ST Dragon	225 FTTC
Speed ball 2	249 FTTC
Stun Runner	245 FTTC
Tom and the ghost	245 FTTC
SWAP	245 FTTC
Wolfpack	295 FTTC
F29	245 FTTC
F19	275 FTTC
M1 Tank platoon	255 FTTC
Dame simulator	235 FTTC
Chess simulator	235 FTTC
Chess champion 2175	290 FTTC
Captive	249 FTTC
SWIV	249 FTTC
The Teller	199 FTTC
Mig 29 Folcrum	NC
Golden axe	245 FTTC
Elvira	299 FTTC

TOURNEZ LA PAGE
ET DECOUVREZ TOUS LES AVANTAGES
DU PRODIPACK EN VOUS ABONNANT
GRATUITEMENT
AU "PETIT GENERALISTE"

SERVICE VPC 10, bd de Strasbourg 75010 PARIS

Si vous ne pouvez vous déplacer ce service est fait pour vous

6 MODES DE RÈGLEMENT AU CHOIX :

1) EN CONTRE-REMBOURSEMENT : vous ne réglez que lors de la livraison de votre matériel. Prévoir 24 F de frais de contre-remboursement lors d'une expédition P&T et 60 F par un transporteur (envoi de plus de 5 kg ou d'une valeur de plus de 2 000 F) en sus du forfait transport. Remplissez le bon de commande ci-dessous. Si vous ne souhaitez pas mutiler votre revue, photocopiez-le ou écrivez-nous sur papier libre ou encore, téléphonez-nous (demander M. BERNARD au (1) 42 06 50 50, postes 436 ou 441).

2) PAR CARTE BLEUE : vous pouvez nous téléphoner et nous commander le matériel désiré, en nous donnant le numéro de votre carte bleue. Vous pouvez également utiliser le bon de commande ci-dessous.

3) PAR CHÈQUE OU CCP : vous remplissez le bon de commande ci-dessous en joignant le règlement du total de votre commande plus le forfait transport correspondant à l'ordre de GÉNÉRAL VIDÉO (voir ci-contre).

4) PAR MANDAT : vous remplissez le bon de commande ci-dessous en précisant que vous désirez régler par mandat postal ou par mandat-lettre. N'omettez pas de compter le forfait transport (voir ci-dessous). A réception de votre mandat, nous expédions votre commande.

5) PAR MINITEL : **TAPEZ 3615 code GVF**, puis choisissez dans notre catalogue Minitel le ou les articles que vous souhaitez commander. Indiquez-nous votre adresse complète, votre numéro de téléphone et votre numéro de Carte Bleue. Si vous souhaitez des renseignements ou une documentation supplémentaire, vous pouvez nous poser vos questions et nous vous répondrons dans les délais les plus courts. Vous pouvez aussi vous amuser sur notre serveur grâce aux concours permanents et gagner de nombreux lots.

6) A CRÉDIT : si vous souhaitez des renseignements détaillés pour l'achat à crédit, vous pouvez joindre Mlle TANIA, Responsable Crédit, au (1) 42 06 50 50, poste 458. Si tout vous paraît clair, vous envoyez le bon de commande ci-dessous en précisant si vous souhaitez verser une partie comptant. Par retour du courrier, nous vous envoyons un dossier de crédit que vous devez nous retourner dûment rempli et signé. Après acceptation de ce dossier, nous vous expédions le matériel. Pièces demandées : pièce d'identité, dernier bulletin de salaire, RIB, quittance EDF ou de loyer informatisée.

• **FORFAIT TRANSPORT** :
20 F pour 2 softs, 35 F pour 3 softs et plus.
50 F pour les accessoires et les disquettes vierges.
60 F pour les imprimantes.
120 F pour les machines.

(Dom-Tom et étranger, nous consulter).
Ce forfait n'est pas cumulable, c'est-à-dire que si vous commandez un micro-ordinateur plus des logiciels, vous ne payez que le forfait transport machine, donc 120 F. Tous nos matériels sont expédiés en recommandé.

ATTENTION : pour toute commande supérieure à 10 000 F, joindre un acompte de 20 % à la commande

BON DE COMMANDE EXPRES

Je, soussigné, déclare commander à GÉNÉRAL, les marchandises et fournitures ci-après désignées pour expédition à mon adresse indiquée ci-contre.

Je choisis de régler par : ☐ Contre-remboursement ☐ Carte Bleue ☐ Chèque/CCP ☐ Mandat ☐ Crédit

* N° de Carte Bleue ______ Date expir. CB ____

N° Carte Aurore ______

NOM ______
Prénom ______
N° ____ Rue ______
Code Postal ______
Ville ______
Tél. ______

ST-02-91

RÉF.	DÉSIGNATION	Pour toute commande de logiciels, précisez le type de la machine et le format de(s) disquette(s) désiré(es)	QUANTITÉ	PRIX UNITAIRE	MONTANTS

Oui, je désire commander le catalogue "LE NOUVEAU GÉNÉRALISTE"* ☐	35F
Oui, je désire recevoir gratuitement "LE PETIT GÉNÉRALISTE"* ☐	Gratuit

BON DE COMMANDE A RETOURNER A GÉNÉRAL, 10 BD DE STRASBOURG, 75010 PARIS

Observations du client : ______

Avez-vous déjà commandé par correspondance chez GÉNÉRAL depuis le 01.09.90
OUI ☐ NON ☐
Date ______

Signature

Pour les mineurs, la signature des parents est obligatoire.

TOTAL COMMANDE	
+ FORFAIT DE PORT	
+ FRAIS CONTRE-REMBOURS.	
TOTAL A RÉGLER	

*Cocher la case de votre choix.

VENTE EN GROS

PLUS DE 6500 ENTREPRISES ET COLLECTIVITES NOUS ONT DEJA FAIT CONFIANCE : VOUS AUSSI, OUVREZ UN COMPTE CHEZ GENERAL

Vous aurez droit à des prix professionels, livraison sous 24 heures, réglement sur relevé de factures.

La vente en gros s'adresse aux entreprises, collectivités, administrations, etc...

Les membres de ces collectivités peuvent se rendre individuellement chez GENERAL, munis d'un justificatif. Il se verront remettre une CARTE COLLECTIVITE qui leur donnera accès aux prix de gros.

Pour toute information, contactez M.N'GOM, M.DELPIT ou Mme LE POULL, tél. 42.06.50.50, télex 214.034, télécopie 42.38.35.60.

Entreprises, nous sommes en mesure de vous proposer les services suivants pour l'acquisitionde votre matériel chez GENERAL :

1°) ouverture d'un compte, après la première commande ;

2°) un financement personnalisé : crédit-bail pour tout matériel d'une valeur supérieure à 10.000 F à condition d'avoir au moins deux exercises d'activité. Ce mode de financement vous permet en outre de récupérer la TVA sur les loyers mensuels.

3°) une formation d'initiation gratuite de votre personnel avec chaque machine vendue (durée 1/2 journée) ;

4°) une formation approfondie, soit dans nos locaux, soit auprès d'instituts spécialisés qui ont notre confiance tel le prestigieux ALTITUDE 21 ;

5°) une assistance téléphonique, sans limitation de temps ;

6°) une maintenance sur site au travers de puissants groupes spécialisés tels AMSTI ou MIS ;

7°) une tarification étudié en fonction des quantités souhaitées et de votre statut d'entreprise.

ETUDIANTS

NOTRE TARIF COLLECTIVITE EST AUSSI POUR VOUS

En dehors des prix promotionnels et des actions temporaires GENERAL, vous avez droit à un tarif spécial sur l'ensemble des produits de notre magasin. L'inscription pour obtenir la CARTE ETUDIANT GENERAL est immédiate et gratuite.

CONDITIONS DE VENTE : ce catalogue n'est valable que pour le mois de sa parution et remplace les précédentes éditions. Produits proposés dans la limite des stocks disponibles. Nos promotions ne sont pas cumulables.



Désormais, la Rubrique Jeux de ST Mag présente ses photos avec un cadre de couleur afin d'identifier la version photographiée :
- cadre rouge : Amiga.
- cadre bleu : ST.

CAR-VUP

CORE **AMIGA / ST**



Après la série des Roger Rabbit, Car Vup nous entraîne encore une fois dans le monde des "Toons". Cette fois-ci, il ne s'agit pas du héros, Roger Rabbit, mais de son vieux pote, Arnie le Taxi. Vous vous rappelez de cette vieille caisse colorée avec un accent new-yorkais à couper au couteau. Donc Arnie va devoir sauver Cartoonland de l'abominable Captain Grim qui ne pense qu'à une chose : détruire tout signe de joie de paix et de tranquillité. Sur ce,



il enlève les amis d'Arnie et Cartoonland se retrouve peuplé de fous en liberté. Vous aurez à nettoyer les huit mondes de Cartoonland, chacun étant composé de six niveaux. L'action commence par le chantier, le niveau fantôme, le jardin, la préhistoire... jusqu'à un niveau nommé "Bizarre" !
Le jeu emprunte un peu le système de Pac-Man, puisqu'il faut rouler sur toutes les plates-formes afin de les désinfecter de l'influence de Captain Grim. Vous devez aussi ramasser tous les objets polluants que vous trouverez, des fruits bonus, et les lettres du mot BONUS et EXTRA. Attention : le temps est limité, et lorsque le chronomètre arrivera à 0, un démon nommé "Turbo" fera son apparition dans le monde et vous poursuivra. A cela, aucune parade, si ce n'est qu'il faut finir le niveau au plus vite. De nombreux gadgets seront à votre disposition (13 exactement), mais notre préféré reste celui qui transforme les ennemis en bonus. Reste un fait important : Arnie ne possède pas de freins, il est donc impossible de s'arrêter (NDLR: judicieuse remarque Jean !) (NDRC : who's that "Jean" fellow ?).
Les graphismes (et la musique) sont dans la lignée de Rainbow Island. La jouabilité est à peu près la même. Le seul défaut notable est que l'on ne peut pas accéder au monde de son choix dès le début, ce qui est regrettable étant donné le nombre de niveaux.

Note: 16 / 20 Jean Delaite



ENCHANTED LAND

THALION **ST**

L'action se déroule dans un petit pays où la joie et la bonne humeur règnent. Je devrais plutôt dire régnaient, car Plogthorn, un vieux sorcier aigri de ne plus avoir de clients décida un jour d'invoquer les esprits du Mal afin de détruire la renommée du Roi. Pour cela, il détruisit l'Ame de la Sagesse. Sans elle, le monde fut plongé dans le Chaos. Un sorcier, moins puissant celui-là, nommé Kurgan fut dépêché sur les lieux afin de rassembler les fragments de l'Ame de la Sagesse. Ce sorcier, c'est vous qui l'incarnerez. Vous devrez parcourir les six régions du monde enchanté avant d'affronter le méchant Plogthorn face à face. Les graphismes sont très beaux et très colorés.





Le scrolling est multidirectionnel mais aussi multiplan, et qui plus est, très fluide. Des tonnes d'obstacles et de monstres terrifiants (escargots, abeilles, puces, araignées...) (NDRC : plus quelques pigistes ?!) se dresseront sur votre chemin. Pour les contrecarrer vous aurez plein d'armes (sous forme de sorts) : bouclier, tir à tête chercheuse, tir rotatif, pistolet. Krugan trouvera aussi des bombes de toutes sortes, mais également des champignons qui augmenteront vos compétences ainsi que des potions. L'écran de jeu est géré de façon à ce qu'il paraisse plein écran. En effet, le bandeau contenant les caractéristiques est si petit (et bien intégré) qu'on ne le voit pratiquement pas (c'est tout de même un peu gênant). Enchanted Land est un jeu d'arcade fort bien réalisé et agréable.

Note: 17 / 20 Jean Delaite

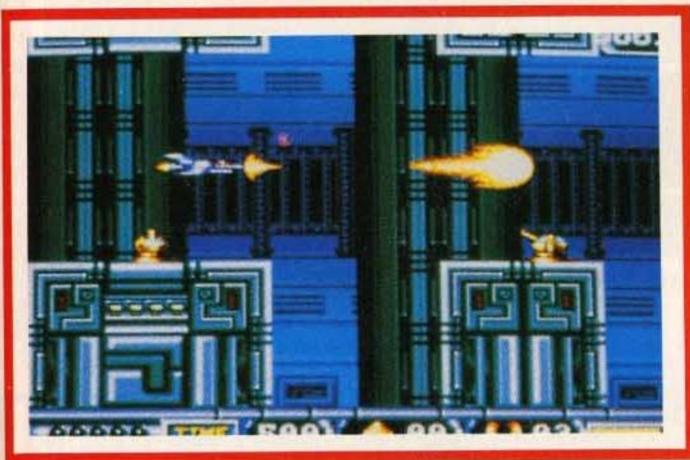
TURRICAN II

RAINBOW ARTS **AMIGA / ST**



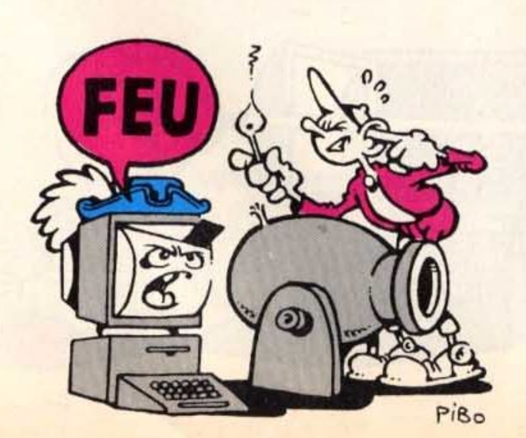
Alors que Turrican sortait sur nos bécanes, les concepteurs s'attaquaient déjà à la suite. Suite fort attendue d'ailleurs, et c'est avec un réel plaisir que j'ai pu y jouer. Sur STE les graphismes sont superbes, mais ce n'est rien comparé à





l'animation du personnage qui évolue avec une grâce digne d'un petit rat de l'opéra (sans le tutu) (NDLR : nous ne devons pas avoir les mêmes valeurs !). Le scénario est typique des kill'em'all : morgull (que vous combattiez dans le premier volet) est encore en train d'envahir une planète (c'est d'ailleurs fait) et c'est à vous d'en chasser ses troupes. Le jeu est composé de huit niveaux dont certains se transformeront en "Shoot'em'Up", car votre personnage dirige un vaisseau à l'intérieur d'un gigantesque complexe fortement armé. Tous les niveaux sont d'ailleurs immenses et non linéaires (vous devez donc vous balader dans tous les sens). D'autres options ont changé : les armes sont plus nombreuses et plus dévastatrices, surtout l'arme suprême qui transforme le robot en un colossal typhon explosif pendant dix secondes. Le tout formant un superjeu très bien construit et, qui plus est, jouable.

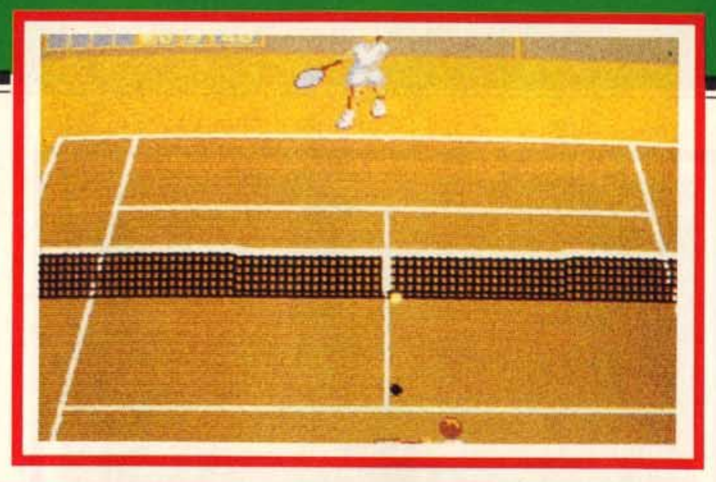
Note: 17 / 20 Jean Delaite





GREAT COURTS II

UBI SOFT **AMIGA / ST**



Personne ne me contredira, si j'écris que Great Courts fit date dans l'histoire des jeux de tennis sur ordinateur. Eh bien, la deuxième édition tient toutes ses promesses en beaucoup plus beau et beaucoup plus complet. Je m'explique : tout d'abord, les graphismes sont vraiment superbes, surtout au niveau des personnages. De plus, l'animation des joueurs ou joueuses est réellement somptueuse, les mouvements sont hyperréalistes et très fluides.

Un autre point fort de Great Courts 2 sont les nombreuses options qui vous sont proposées. Vous avez la possibilité de jouer un homme, une femme, un double masculin ou féminin, un mixte. Mais là où Great Courts 2 est génial, c'est que l'on peut y jouer à quatre simultanément (avec un doubleur de joystick). Pour ce qui est des coups, le timing est de rigueur mais il n'y a guère de changements. Par contre, en mode débutant ce sera plus simple, puisque l'ordinateur gère les déplacements. De plus chaque joueur est personnalisable, puisque vous devez lui attribuer des points dans certaines compétences. Vous aurez par exemple, un spécialiste du lobe ou du passing shot, ce qui aura pour but d'augmenter vos chances de réussir le coup, mais aussi de surprendre l'adversaire. En conclusion, l'attente fut récompensée et c'est avec un produit fort qu'Ubi Soft réattaque le marché du Tennis.

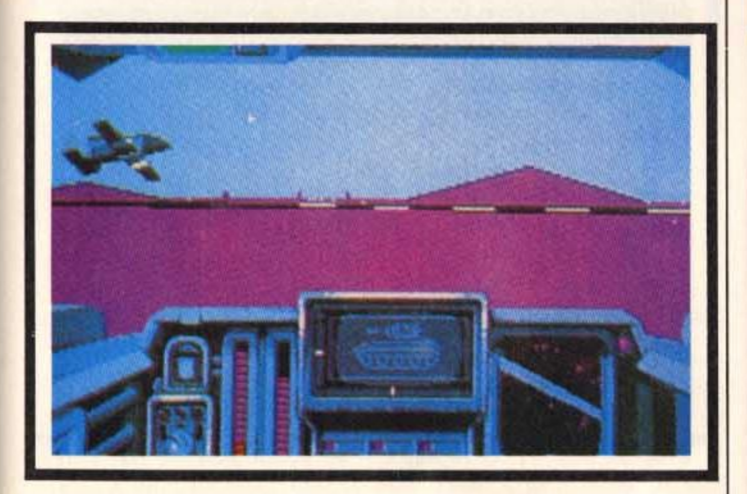
Note: 18 / 20 Jean Delaite

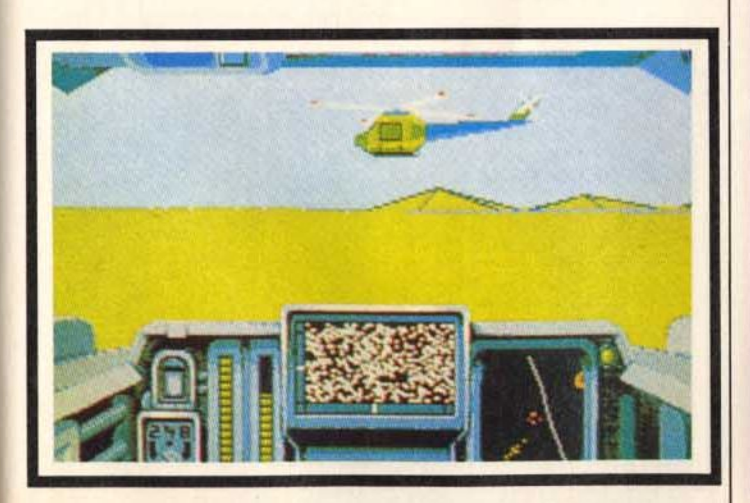


BATTLE COMMAND

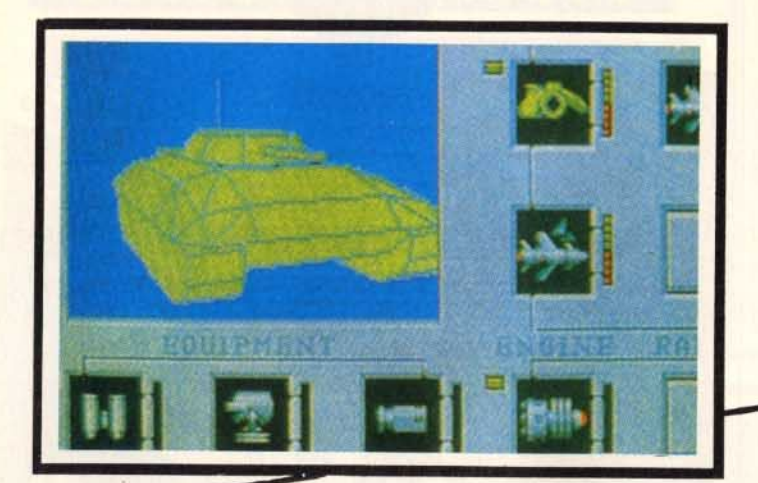
OCEAN **AMIGA / ST**

Je pense que vous vous rappelez de l'excellent Carrier Command qui vous mettait aux commandes d'un super porte-avions futuriste. Eh bien Battle Command pourrait pratiquement être considéré comme la suite de ce dernier puisqu'il est réalisé par la même équipe. Battle Command se déroule sur la terre ferme et aux commandes d'un superchar, le tout dans un univers 3D. Le jeu n'est pas sans rappeler M4 Sherman (Loriciel), mais les missions sont tout de même beaucoup plus intéressantes. Missions qui commencent par une sélection de votre armement : vous avez le choix entre quatre types de missiles, plus des leurres. Puis vient l'ordre de mission avec votre objectif représenté sur une carte. Sur cette même carte figurent votre point de largage (le tank est parachuté d'un avion), votre objectif, puis le point de récupération. Les missions proposées vous entraîneront vers des zones industrielles à détruire, récupérer un satellite, détruire un train, escortes, reconnaissances... (NDRC : toutes occupations déjà trop familières...) Il y en a seize en tout. Les ennemis sont aussi très bien armés et qui plus est, très nombreux : tanks, hélicoptères, avions (NDRC : le type de conflit se précise).









Les décors, quant à eux, ne sont pas formidables, du moins de loin, car si vous vous rapprochez des maisons, vous verrez les fenêtres, les portes et autres détails. Le scrolling est rapide et fluide que ce soit en vue intérieure (du tank) ou extérieure. En fin de compte, Battle Command est plein de bons éléments qui feront la joie des amateurs du genre simulation-action.

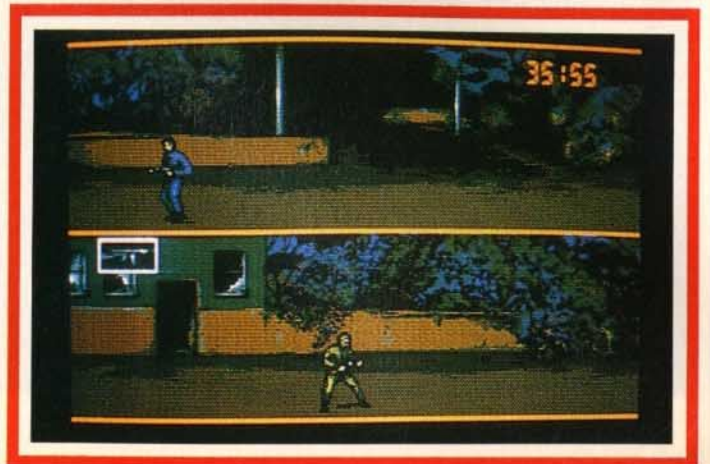
Note: 16 / 20 Jean Delaite

LES PREVIEWS

SNOW BROTHERS (OCEAN / Amiga / ST)

La suite de Bubble Bobble sera bientôt disponible sur nos micros. Cette fois-ci, il s'agit d'immobiliser vos adversaires à l'aide de boules de neige pour ensuite les envoyer contre les murs ou d'autres ennemis. Aussi jouable et passionnant que Bubble Bobble, Snow Brothers devrait connaître, au moins, le même succès.





ALCATRAZ (INFOGRAMES / Amiga / ST)

Fortement inspiré d'Hostages, Alcatraz est un jeu d'arcade à plusieurs facettes. Avant de détruire la bande de trafiquants de drogue ayant investi l'endroit et de délivrer leurs otages, vous devrez entrer par surprise, déambuler dans un labyrinthe de cellules pour enfin affronter l'ennemi. Sortie prévue en mars.





HAMMER IN HARRY (OCEAN / Amiga / ST)

Excellent jeu d'arcade, Hammer in Harry est un jeu de tableau dans lequel un personnage armé d'un marteau part en guerre contre des promoteurs immobiliers qui n'ont pas hésité à raser sa maison pour construire un nouvel édifice. La réalisation du jeu d'arcade est somptueuse, gageons qu'il en sera de même pour la version micro. Prévue pour septembre 91.



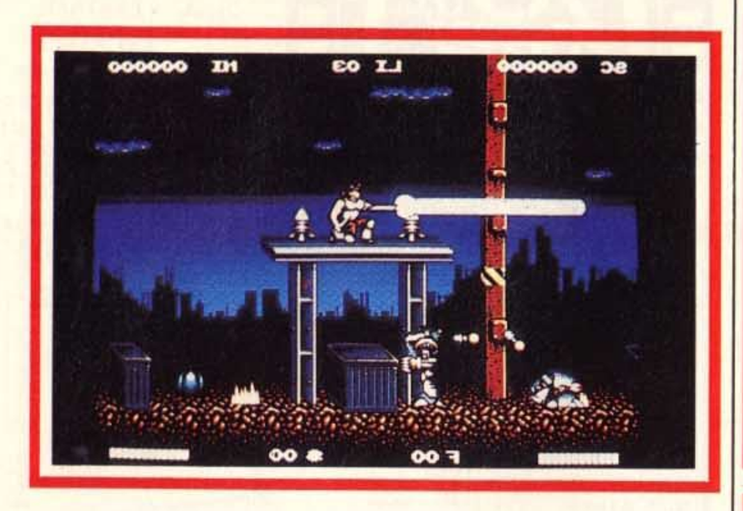


LIQUID KIDS (OCEAN / Amiga / ST)

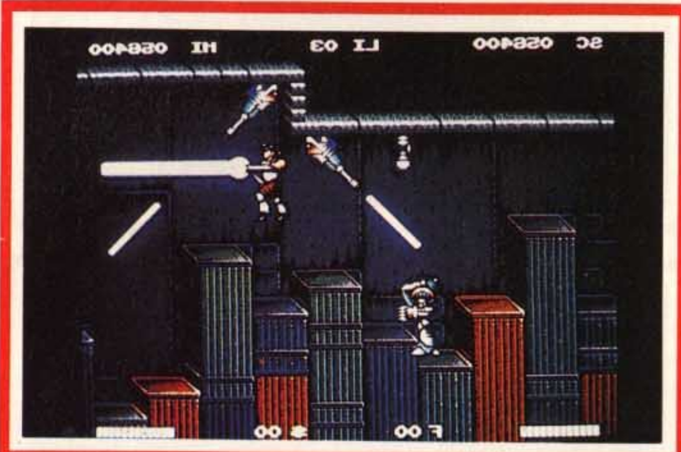
Réalisé par les auteurs de New Zealand Story, Liquid Kids en est la suite logique. Encore plus dément, plus beau, plus jouable, ce jeu fera sûrement un carton auprès du public. Prévu également pour septembre 91.







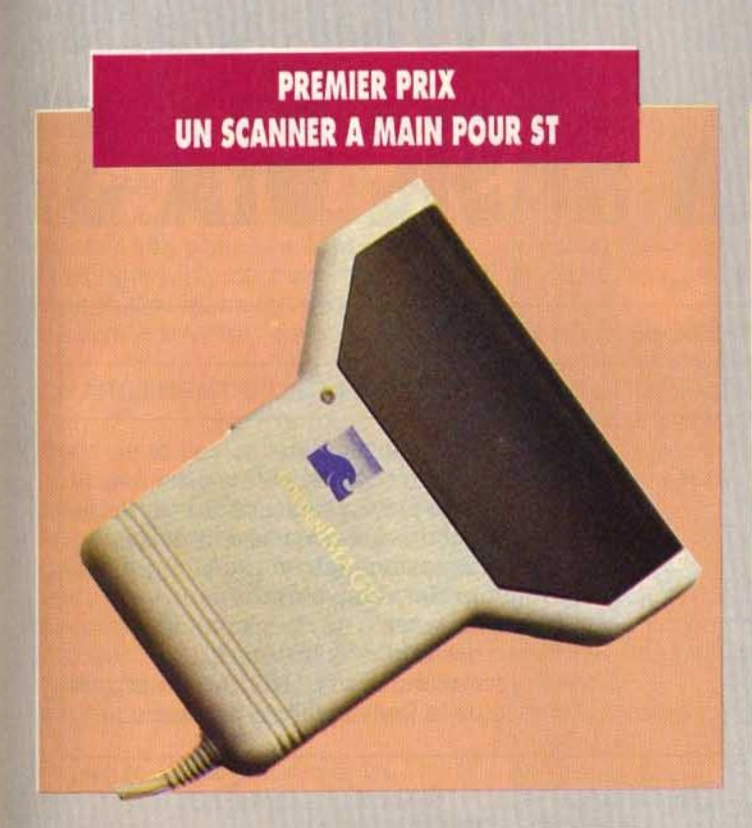




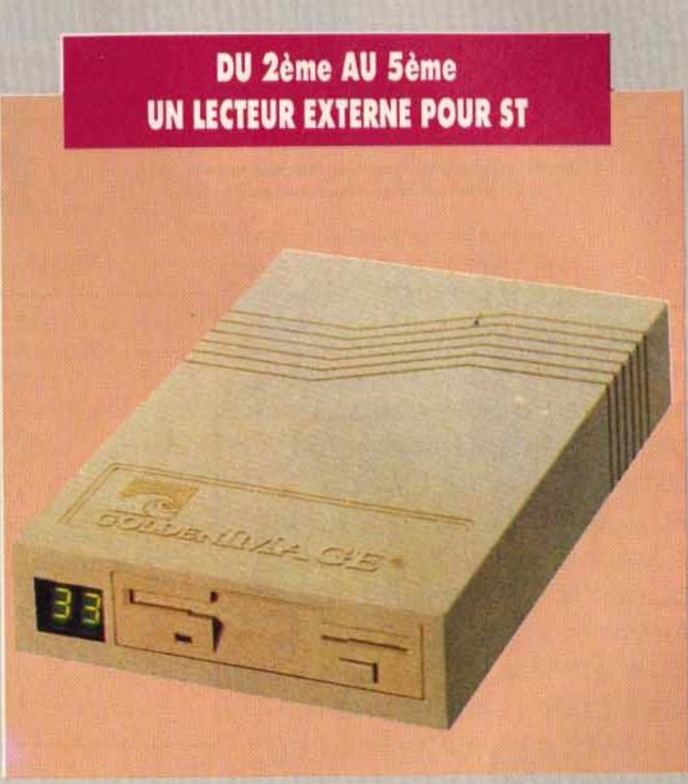
SWITCHBLADE 2 (GREMLIN / Amiga / ST)
Le phénoménal succès du premier épisode appelait forcément une suite, c'est maintenant chose faite avec la suite des aventures de notre personnage. Ce dernier a considérablement évolué et montre dorénavant une apparence humaine. Le jeu est toujours aussi immense et passionnant, mais la réalisation s'est grandement améliorée avec des graphismes superbes et une animation excellente. Test normalement le mois prochain.

QUIZZ IMAGINE'S - 3615 STMAG

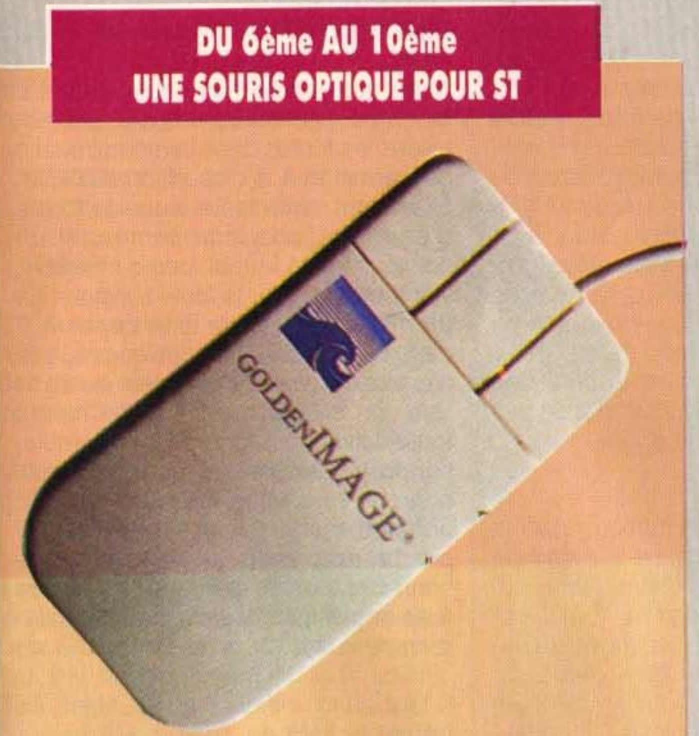
PREMIER PRIX
UN SCANNER A MAIN POUR ST



DU 2ème AU 5ème
UN LECTEUR EXTERNE POUR ST



DU 6ème AU 10ème
UNE SOURIS OPTIQUE POUR ST



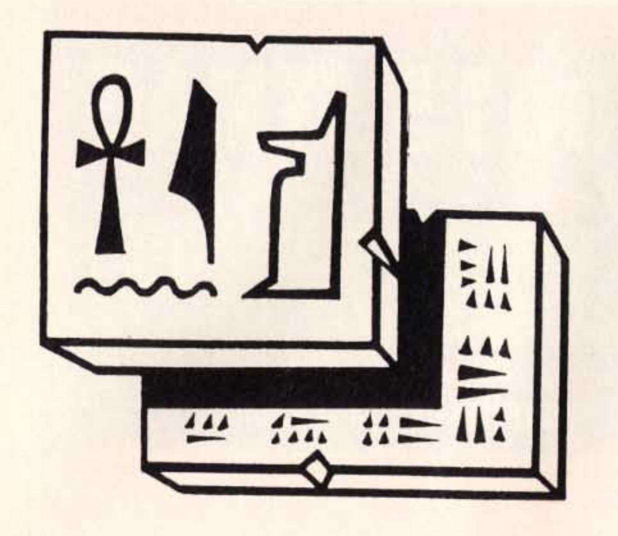
DIX PRIX EN TOUT!
COMMENT PEUT-ON PARTICIPER?!

Répondez sur le 3615 STMAG aux questions concoctées par la Rédaction de ST Mag.

Connaissez-vous suffisamment bien le ST pour avoir une chance de faire partie des 10 meilleurs joueurs?

Un grand merci à IMAGINE'S pour ces dix lots GOLDEN IMAGE!

IMAGINE'S
27-41 boulevard Louise Michel
92230 GENNEVILLIERS
Tél: 47 91 06 25



UTILISER LES FONTES GEM EN BASIC GfA 3

La possibilité d'utiliser des jeux de caractères, différents de la fonte système du ST, est un problème souvent évoqué dans le rubrique GfA du 3615 STMAG. Une solution serait de travailler avec le brave GDOS, mais sa lourdeur d'installation a de quoi rebuter. Il m'a donc paru intéressant de mettre au point des procédures en GfA 3 pour réaliser cette fonction. Et pendant qu'on y est, pourquoi ne pas améliorer les choses, en particulier en gérant les fontes individuellement ou en les intégrant dans des INLINE, par exemple ?

TABLEAU 1 : En-tête des fichiers fontes

Offset	Taille	Fonction
00	Mot	Numéro identifiant la fonte
04	Chaîne	Nom de la fonte (terminé par un octet nul)
36	Mot	Code ASCII du premier caractère de la fonte
38	Mot	Code ASCII du dernier caractère de la fonte
40	Mot	Hauteur des caractères en pixels
66	Mot	Indicateurs binaires (voir texte)
72	Mot long	Adresse de la table de décalage
76	Mot long	Adresse des données "bit-map"
82	Mot	Hauteur des cellules encadrant les caractères en pixels
84	Mot long	Adresse de la fonte suivante en mémoire

LE FORMAT DES FICHIERS FNT

Les fichiers fontes utilisés par le GEM comportent dans un en-tête toute une série d'informations nécessaires à l'écriture graphique. Vous pouvez trouver le détail de ces données dans des livres tels que "La Bible du ST". Les principales valeurs, qui seront utilisées par la suite, sont rappelées dans le tableau 1 ci-contre.

Pour commencer à corser les choses, cet en-tête est toujours au format Intel, c'est-à-dire que chaque mot est codé avec l'octet de poids faible précédant l'octet de poids fort. La table de décalage (dont le rôle est d'indiquer où se trouve le dessin bit-map de chaque caractère dans le fichier FNT), est elle aussi systématiquement à ce format. Par contre, les données graphiques définissant les caractères proprement dits, peuvent être soit en mode Intel, soit en mode normal (pour le ST...) Motorola : c'est le bit 2 des indicateurs binaires qui permet de distinguer entre ces deux cas.

Le bit 3, quant à lui, indique que la fonte utilisée est non-proportionnelle (tous ses caractères ont la même largeur, comme les 'system fonts' par exemple). Lorsqu'il reconnaît une telle fonte, le GEM utilise des algorithmes plus rapides pour dessiner les caractères, d'où l'intérêt de la chose.

ORGANISATION DES FONTES EN MÉMOIRE

Avant de charger un fichier fonte, il faut savoir comment doivent être installées les fontes dans la mémoire, afin de les mettre à la disposition du GEM. Elles sont organisées sous la forme d'une liste, chacune contenant un pointeur (dans le mot long d'offset 84, voir tableau 1) sur la fonte suivante. La dernière fonte de la liste contient 0 dans ce mot long, pour indiquer qu'elle n'a pas de suivante. Au sein de cette liste, les diverses tailles d'une même fonte doivent être groupées ensemble, par ordre de taille croissante. Lorsqu'il cherche une fonte donnée, le GEM parcourt tranquillement cette liste de proche en proche, jusqu'à trouver la bonne fonte avec la bonne taille (ou la taille la plus proche), ou bien atteindre la fin de la liste.

Il faut aussi indiquer au GEM où se trouve la liste en question. Il existe

pour cela quatre pointeurs, regroupés dans une zone de mémoire répondant au doux nom de FONT_RING : le premier pointe sur la "6x6 system font", le second sur la liste des fontes intégrées dans la ROM du ST (seulement les fameuses "system font", hélas...), et le troisième sur la liste des fontes supplémentaires chargées ultérieurement. Le quatrième pointeur reste à 0 et marque la fin du FONT_RING.

Avec le GfA, on peut obtenir l'adresse du FONT_RING par L~A-456, et les pointeurs sur les trois listes : {L~A-456} pour la fonte 6x6, {L~A-452} pour les fontes intégrées, et {L~A-448} pour les fontes GEM. Normalement, ce dernier pointeur vaut 0, tant qu'on n'a pas enregistré de nouvelle fonte.

CHARGEMENT D'UNE FONTE

Forts de toutes ces précieuses indications, nous allons pouvoir nous attaquer au vif du sujet, c'est-à-dire le chargement et l'enregistrement d'une fonte. Le moment est donc venu de vous reporter aux listings accompagnant cet article, en commençant par la procédure nommée 'load_font' (nom original s'il en est !). La syntaxe de l'appel est :

```
@load_font(font_pter%,font_id&,
           font_height&,font_name$)
```

Si la fonte en question se trouve dans un fichier, font_pter% doit contenir un pointeur sur le nom de ce fichier. Par exemple :

```
FILESELECT "\*.FNT","",font_file$
IF LEN(font_file$)>0 THEN
   @load_font(*font_file$, etc...)
ENDIF
```

En revanche, si vous avez mis la fonte dans un INLINE, font_pter% doit contenir un pointeur sur l'adresse du INLINE. Par exemple :

```
INLINE font_addr%, taille fichier fonte
@load_font(*font_addr%, etc...)
```

Le fait de passer un pointeur sur la variable, plutôt que la valeur de la variable, permet d'écrire une seule procédure qui s'adapte à chaque cas par le truchement de la fonction TYPE du GfA.

```
REM    *-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
REM    | Procédures de gestion des fontes GEM en GFA 3, par NCC1701 |
REM    *-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
REM Variables globales réservées (ne doivent pas être modifiées par ailleurs) :
REM font_base%     : adresse de la première fonte de la liste
REM scratch_pter% : adresse du tampon graphique pour calcul des caractères
REM scratch_size&   : taille courante du tampon graphique
REM scratch_offset& : offset dans le tampon graphique pour les effets
REM ********Chargement/Enregistrement d'une fonte GEM
REM @load_font( *var$ contenant le nom du fichier fonte à charger ou
REM             *var% contenant l'adresse du INLINE où est la fonte,
REM   variable & renvoyant l'identification de la fonte ou code err<0,
REM   variable & renvoyant la hauteur des caractères de la fonte,
REM   variable $ renvoyant le nom de la fonte)
'
PROCEDURE load_font(font_pter%,VAR font_id&,font_height&,font_name$)
  LOCAL font_length%,font_succ%,font_addr%
  cancel_fonts                         ! Annule les fontes actuelles
  SELECT TYPE(font_pter%)              ! Selon le type du pointeur reçu,
  CASE 1                               ! Fonte en fichier,
    font_name$=LEFT$(CHAR{{font_pter%}},INT{font_pter%+4})
    OPEN "I",#99,font_name$            ! ouvre le fichier en lecture
    font_length%=LOF(#99)              ! taille du fichier
    font_addr%=MALLOC(font_pred%)      ! réservation de mémoire
    IF font_addr%=0 THEN               ! si erreur,
      font_id&=-39                     ! renvoie code erreur -39
    ELSE
      BGET #99,font_addr%,font_length% ! charge le fichier fonte
    ENDIF
    CLOSE #99                          ! fermeture du fichier
  CASE 2                               ! Fonte en INLINE,
    font_addr%={font_pter%}            ! récupère l'adresse de la fonte
  DEFAULT
    font_id&=-1                        ! Type invalide, code erreur -1
  ENDSELECT
  IF font_id&>=0               ! SI pas d'erreur de type ou de lecture,
    font_init(font_addr%)              ! Initialisation de la fonte
    font_id&=INT{font_addr%}           ! N• identification de la fonte
    font_name$=CHAR{font_addr%+4}      ! Nom de la fonte
    font_height&=INT{font_addr%+40}    ! Hauteur des caractères
    ' *** Vérifie si la fonte ne figure pas déjà dans la liste
    font_seek(font_id&,font_height&,font_pred%,font_pter%,font_succ%)
    ' *** Vérifie également qu'on peut réserver un tampon
    IF font_pter%=0 AND MALLOC(-1)>MAX(scratch_size&,INT{font_addr%+70})
      IF font_pred%=0                  ! Insertion comme première fonte
        {font_addr%+84}=font_base%
        font_base%=font_addr%
      ELSE IF font_succ%=0             ! Insertion comme dernière fonte
        {font_pred%+84}=font_addr%
        {font_addr%+84}=0
      ELSE                             ! Insertion au sein de la liste
        {font_addr%+84}=font_succ%
        {font_pred%+84}=font_addr%
      ENDIF
      {font_addr%+72}=font_addr%+{font_addr%+72} ! Reloge pointeur décalage
      {font_addr%+76}=font_addr%+{font_addr%+76} ! Reloge pointeur données
      scratch_size&=MAX(scratch_size&,INT{font_addr%+70}) ! Taille scratch
      scratch_offset&=MAX(scratch_offset&,INT{font_addr%+68}) ! Offset scratch
    ELSE
      ~MFREE(font_addr%)               ! Si erreur, libère la mémoire
      font_id&=-2                      ! code erreur -2
    ENDIF
```



Au retour de la procédure, font_id& contiendra en principe le numéro d'identification de la fonte (très important pour la suite), font_height& la hauteur des caractères et font_name$ le nom de la fonte. En cas de problème, font_id& contiendra un numéro d'erreur négatif : -1 si vous avez passé un mauvais pointeur dans font_pter%, -2 si la fonte est déjà dans la liste du GEM (pas la peine d'encombrer la précieuse mémoire de votre ST !), -39 si la mémoire disponible est insuffisante pour charger la fonte (cas d'un fichier sur disque).

Un gros travail consiste alors à initialiser la fonte. Cet exploit est assuré par la procédure font_init, qui se charge des tâches suivantes : mise au format MOTOROLA de l'en-tête et de la table de décalage, calcul de la taille du tampon nécessaire au GEM pour dessiner les caractères, et positionnement correct des indicateurs binaires. A ce sujet, il est amusant de remarquer que le bit 2 (qui indique le format INTEL ou MOTOROLA), est utilisé de façon exactement inverse par le GDOS et la ROM : les fontes GEM étant conçues pour fonctionner avec le GDOS, il convient donc d'inverser ce bit (d'où le XOR 4 dans la procédure font_init), sous peine d'obtenir d'infâmes gribouillis en lieu et place des jolis caractères prévus !

Il ne reste plus qu'à insérer la fonte dans la liste du GEM. Pour cela, on utilise la procédure font_seek, qui recherche une fonte donnée dans la liste actuelle : on obtient en retour l'adresse de la fonte cherchée (0 si elle ne figure pas dans la liste, cas qui nous intéresse ici), celle de la précédente (ou 0 si c'est la première de la liste) et celle de la suivante (ou 0 si c'est la dernière). Grâce à ces trois valeurs, il est possible de positionner les pointeurs de chaînage des fontes.

Finalement, l'appel à VST_LOAD_FONTS indique au GEM où se trouve la liste des fontes. Il parcourt alors la liste, se charge de la conversion éventuelle des données au format MOTOROLA (merci GEM !), et compte le nombre de jeux de caractères différents dans la liste. Et voilà, c'est fait, vous avez à votre disposition une nouvelle fonte !

```
  ENDIF
  record_fonts                           ! Enregistre la nouvelle liste
RETURN
REM ********
REM Suppression de fonte(s) de la liste du GEM
REM @unload_font(<identificateur de la fonte ou 0 pour toutes les fontes>,
REM              <hauteur des caractères ou 0 pour toutes les hauteurs>)
'
PROCEDURE unload_font(font_id&,font_height&)
  LOCAL font_pred%,font_succ%,font_addr%
  cancel_fonts                           ! Annule la liste des fontes
  IF font_id&>0                          ! Si un seul groupe de fonte,
    DO                                   ! Tant qu'on trouve ce numéro,
      font_seek(font_id&,font_height&,font_pred%,font_addr%,font_succ%)
      EXIT IF font_addr%=0
      IF font_pred%=0                    ! Suppression de la première fonte
        font_base%=font_succ%
      ELSE IF font_succ%=0               ! Suppression de la dernière fonte
        {font_pred%+84}=0
      ELSE                               ! Suppression au sein de la liste
        {font_pred%+84}=font_succ%
      ENDIF
      {font_addr%+72}={font_addr%+72}-font_addr%  ! Déloge table décalage
      {font_addr%+76}={font_addr%+76}-font_addr%  ! Déloge table données
      ~MFREE(font_addr%)                 ! Libère le bloc-mémoire
    LOOP
  ELSE                                   ! Si toutes les fontes,
    font_addr%=font_base%                ! commence sur la première fonte
    WHILE font_addr%<>0                  ! tant qu'il reste des fontes,
      {font_addr%+72}={font_addr%+72}-font_addr%  ! Déloge table décalage
      {font_addr%+76}={font_addr%+76}-font_addr%  ! Déloge table données
      ~MFREE(font_addr%)                 ! Libère le bloc-mémoire
      font_addr%={font_addr%+84}         ! et passe à la fonte suivante
    WEND
    font_base%=0                         ! RAZ du pointeur liste des fontes
  ENDIF
  scratch_size&=0                        ! Détermine la taille maximum
  scratch_offset&=0                      ! pour le tampon graphique parmi
  font_addr%=font_base%                  ! les fontes restant dans la liste
  WHILE font_addr%<>0
    scratch_size&=MAX(scratch_size&,INT{font_addr%+70})
    scratch_offset&=MAX(scratch_offset&,INT{font_addr%+68})
    font_addr%={font_addr%+84}
  WEND
  record_fonts                     ! Enregistre nouvelle liste de fontes
RETURN
REM ********
REM Changement fonte texte du VDI (alpha-mode), utilisée par ex. par PRINT, le
REM BIOS, les ESCAPE du VDI...). Les fontes utilisées doivent IMPERATIVEMENT
REM commencer au caractère ASCII de code 0, être non-proportionnelles et
REM de largeur 8 ! Par contre, la hauteur des caractères n'est pas limitée.
REM def_alpha_font(<identificateur de la fonte>, (1=fonte système)
REM                <hauteur des caractères>)
'
PROCEDURE def_alpha_font(font_id&,font_height&)
  LOCAL font_pred%,font_addr%,font_succ%  ' Rech.fonte ds liste des fontes GEM
  font_seek(font_id&,font_height&,font_pred%,font_addr%,font_succ%)
  ' Si la fonte existe, vérifie aussi les conditions du mode 'alpha'
  IF font_addr%<>0
    IF INT{font_addr%+36}=0 AND INT{font_addr%+52}=8 AND BTST(INT{font_addr%+66},3)
      {INTIN}=font_addr%                ! Adresse de la fonte dans INTIN
      VDISYS 5,0,0,102                  ! Change la fonte du VDI (ESC 102)
```

```
    ENDIF
  ENDIF
RETURN
REM Changement de la fonte graphique de l'AES (graf-mode). Ceci permet d'utiliser
REM des fontes différentes dans menus, boites d'alerte et formulaires du GEM.
REM def_graf_font(<identificateur de la fonte ou -1>, (1=fonte système)
REM              <hauteur des caractères ou -1>,
REM              <flags d'attributs d'écriture ou -1>,
REM              <couleur d'écriture ou -1>)
PROCEDURE def_graf_font(font_id&,font_height&,font_attr&,font_color&)
  V~H=GRAF_HANDLE()
  IF font_id&>=0                          ! Si changement de fonte,
    INTIN(0)=font_id&
    VDISYS 21                             ! SET TEXT FACE
  ENDIF
  IF font_height&>=0                      ! Si changement de taille,
    PTSIN(1)=font_height&                 ! ( Prudence... )
    CONTRL(1)=1
    VDISYS 12                             ! SET CHARACTER HEIGHT
  ENDIF
  IF font_attr&>=0                        ! Si changement d'attributs,
    INTIN(0)=font_attr&
    VDISYS 106                            ! SET GRAPHIC TEXT EFFECTS
  ENDIF
  IF font_color&>=0                       ! SI changement de couleur,
    INTIN(0)=font_color&
    VDISYS 22                             ! SET GRAPHIC TEXT COLOR
  ENDIF
  V~H=-1                                  ! Replace le handle écran du GFA
RETURN
REM ***********************************************************
REM Procédures pour annuler/enregistrer les fontes pour le GEM
REM ***********************************************************
PROCEDURE cancel_fonts
  IF font_base%<>0 THEN             ! Si la liste des fontes n'est pas vide,
    ~MFREE(scratch_pter%)           ! Libère la mémoire du tampon graphique
    V~H=GRAF_HANDLE()               ! Handle écran de l'AES
    ~VST_UNLOAD_FONTS(0)            ! Efface les fontes
    V~H=-1                          ! Handle écran virtuel du GFA
    ~VST_UNLOAD_FONTS(0)            ! Efface les fontes
  ENDIF
RETURN
PROCEDURE record_fonts
  IF font_base%<>0 THEN                   ! S'il y a des fontes dans la liste,
    scratch_pter%=MALLOC(scratch_size&) ! Réserve mémoire du tampon graphique
    {CONTRL+14}=scratch_pter%             ! CONTRL(7/8)  =adresse tampon graphique
    INT{CONTRL+18}=scratch_offset&        ! CONTRL(9)    =offset graphique
    {CONTRL+20}=font_base%                ! CONTRL(10/11)=adresse première fonte
    V~H=GRAF_HANDLE()                     ! handle écran AES
    ~VST_LOAD_FONTS(0)                    ! Enregistre les fontes pour l'AES
    V~H=-1                                ! handle écran virtuel GFA
    ~VST_LOAD_FONTS(0)                    ! Enregistre les fontes pour le GFA
  ENDIF
RETURN
REM ***************************************************************************
REM Initialisation d'une fonte si nécessaire : Mise au format MOTOROLA de l'en-
REM tête et de la table de décalage. Positionnement correct des "flag-bits" de
REM la fonte. Calcul de la taille du tampon graphique requis
REM font_init(<adresse de la fonte en mémoire>)
PROCEDURE font_init(font_addr%)
  LOCAL font_width&,font_offset&,font_table%,font_count&
  IF NOT BTST(INT{font_addr%+66},15)    ! SI fonte pas encore initialisée,
```

UTILISATIONS D'UNE FONTE

Les procédures proposées permettent trois usages différents des fontes GEM :

a) L'utilisation pour les sorties de textes graphiques avec la commande TEXT du GfA : il suffit de sélectionner la fonte voulue avec la commande DEFTEXT du GfA. C'est le numéro d'identification de la fonte qui doit être indiqué, il est donc important de le mémoriser après l'appel à load_font.

b) Le changement de la fonte alphanumérique du VDI, qui est utilisée par l'émulateur VT52 du TOS (PRINT en GfA). Certaines restrictions sont toutefois à considérer : la fonte doit commencer au caractère de code 0, être non-proportionnelle et de largeur 8 (on aura reconnu un portrait-robot des fontes intégrées au ST...). La procédure def_alpha_font vérifie donc ces conditions avant de changer la fonte par la fonction ESCAPE 102 du VDI.

c) Le changement de la fonte utilisée par l'AES dans les menus, les boîtes d'alerte et les formulaires. Bien que, théoriquement, n'importe quelle fonte graphique puisse être utilisée, il est sage de se restreindre à des fontes ayant des tailles inférieures ou égales à la fonte système. En effet, l'AES base tous ses calculs d'affichage de texte sur la taille de la fonte système : en utilisant des caractères trop grands, on risque de le faire paniquer et il n'apprécie pas du tout ! Le procédure def_graf_font permet aussi de choisir des effets d'affichage (gras, italique, etc.) et la couleur.

SUPPRESSION D'UNE FONTE

Si vous n'avez plus besoin d'une ou plusieurs fontes, il est possible de les supprimer de la liste du GEM au moyen de l'instruction :

```
@unload_font(font_id&,font_height&)
```

Font_id& indique le numéro de la fonte à supprimer, et font_heigth& la taille d'icelle. On peut ainsi supprimer individuellement les fontes. Le bloc de



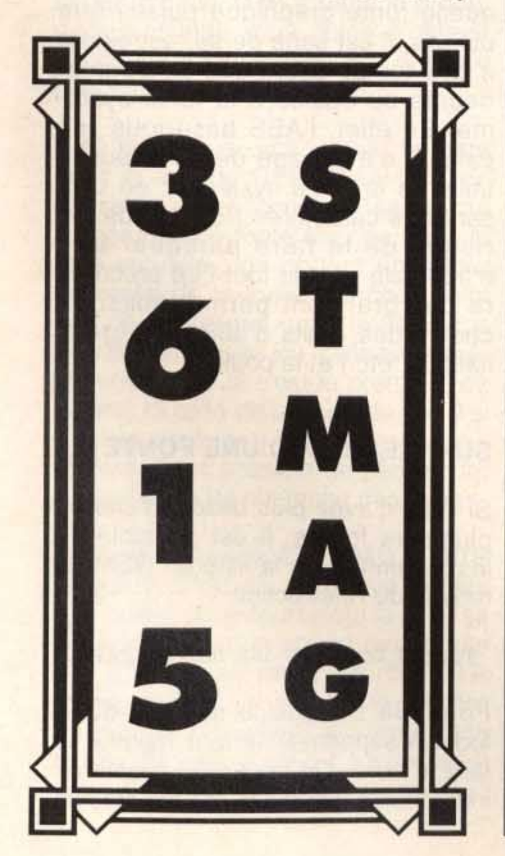
mémoire réservé est libéré, si la fonte avait été chargée depuis un disque.

L'instruction unload_font(font_id&,0) supprime toutes les tailles de la fonte font_id& présentes dans la liste, et unload_font(0,0) supprime purement et simplement TOUTES les fontes de la liste du GEM. Mettez cette instruction dans votre programme avant le retour à l'éditeur ou au bureau, afin de libérer la mémoire utilisée. Pensez également à réactiver la fonte système, dont le numéro d'identification est 1.

REMARQUES FINALES

Pour fonctionner correctement, les procédures proposées ont besoin de quatre variables globales réservées (voir début du listing). Vous ne devez pas modifier le contenu de ces variables dans le corps de votre programme, sous peine d'attirer sur vous des effets indésirables et incontrôlés. Si vous rencontrez des problèmes dans l'utilisation des ces procédures, je m'efforcerai de répondre à vos questions en bal NCC1701 sur le 3615 STMAG.

Emmanuel Talmy




```
    ' *** Mise au format MOTOROLA de l'en-tête de la fonte ***
    INT{font_addr%}=ROR&(INT{font_addr%},8)       ! Identification fonte
    INT{font_addr%+2}=ROR&(INT{font_addr%+2},8)  ! Taille fonte en points
    FOR font_offset&=36 TO 82 STEP 2             ! Reste de l'en-tête
      INT{font_addr%+font_offset&}=ROR&(INT{font_addr%+font_offset&},8)
    NEXT font_offset&
    {font_addr%+72}=SWAP({font_addr%+72})         ! Table de décalage
    {font_addr%+76}=SWAP({font_addr%+76})         ! Table de données
    ' *** Mise au format MOTOROLA de la table de décalage ***
    font_count&=2*SUCC(INT{font_addr%+38}-INT{font_addr%+36})
    font_table%=font_addr%+{font_addr%+72}
    FOR font_offset&=0 TO font_count& STEP 2
      INT{font_table%+font_offset&}=ROR&(INT{font_table%+font_offset&},8)
    NEXT font_offset&
    font_width&=INT{font_addr%+52}        ! Teste si fonte non-proportionnelle
    FOR font_offset&=2 TO font_count& STEP 2
      EXIT IF (INT{font_table%+font_offset&}-INT{font_table%+font_offset&-2})<>font_width&
    NEXT font_offset&
    IF font_offset&>font_count& THEN
      INT{font_addr%+66}=BSET(INT{font_addr%+66},3)
    ENDIF  ! Puis vient calcul taille du tampon pour effets de texte graphique
    w&=INT{font_addr%+52}+INT{font_addr%+54}+INT{font_addr%+56}
    h&=INT{font_addr%+82}
    INT{font_addr%+68}=MAX(256,4*MAX(MUL(h&,DIV(ADD(w&,15),16)),MUL(w&,DIV(ADD(h&,15),16))))
    w&=2*SUCC(w&)
    l&=2*SUCC(h&)
    INT{font_addr%+70}=MAX(512,INT{font_addr%+68}+2*
    MAX(MUL(w&,SUCC(DIV(h&,16))),MUL(h&,SUCC(DIV(w&,16))))
+2*SUCC(2*MAX(SUCC(w&\16),SUCC(h&\16))))
    ' *** Marquage fonte initialisée, inversion du bit de format INTEL ***
    INT{font_addr%+66}=BSET(INT{font_addr%+66} AND &HFFFC XOR &H4,15)
  ENDIF
RETURN
REM Procédure de recherche d'une fonte dans la liste :
REM font_seek( <identificateur de la fonte>,  ( 1=fonte système )
REM        <hauteur des caractères ou 0 pour la première fonte font_id&>,
REM        variable % renvoyant l'adresse de la fonte précédente ou 0,
REM        variable % renvoyant l'adresse de la fonte cherchée ou 0,
REM        variable % renvoyant l'adresse de la fonte suivante ou 0)
PROCEDURE font_seek(font_id&,font_height&,VAR font_pred%,font_addr%,font_succ%)
  LOCAL font_id%,font_id0% ! Regroupement des deux critères de recherche
  font_id%=SWAP(font_id&)-font_height&*(font_height&>0)! dans 1 seule variable
  font_pred%=0
  IF font_id&=1                         ! Si on cherche une fonte système,
    font_addr%={L~A-452}                ! adresse de la liste système
  ELSE                                  ! Sinon,
    font_addr%=font_base%               ! adresse de la liste GEM
  ENDIF
  WHILE font_addr%<>0                   ! Parcourt la liste des fontes
    font_id0%=SWAP(INT{font_addr%})-INT{font_addr%+40}*(font_height&>0)
    EXIT IF font_id%<=font_id0%         ! jusqu'à trouver critère requis
    font_pred%=font_addr%               ! ou la fin de la liste
    font_addr%={font_addr%+84}
  WEND
  IF font_id%=font_id0% AND font_addr%<>0      ! Si fonte trouvée,
    font_succ%={font_addr%+84}                 ! renvoie son adresse
  ELSE
    font_succ%=font_addr%
    font_addr%=0                               ! Sinon, renvoie 0
  ENDIF
RETURN
```



Tonton Billy T. continue à vous guider dans l'exploration de la simulation des phénomènes aléatoires. Armé d'un Générateur de Nombres Aléatoires (GNA) bien affûté, et prenant comme exemples des éléments de programmation d'un jeu (imaginaire) d'exploration spatiale, Billy T. vous emmène aujourd'hui examiner les distributions aléatoires et leur programmation.

RAPPEL

Pour ceux qui prendraient le train en marche, il est recommandé de lire les volets précédents dans les numéros 45 et 46 de ST Mag. Rappelons seulement que de nombreux GNA existent dans la littérature informatique et que la plupart sont des aberrations, voire des monstres bugués jusqu'à l'os. Le GNA préconisé (et looonguement testé) par Billy T., votre serviteur, consiste en un algorithme très simple : on fixe pour un entier z une valeur initiale, nommée semence, et chaque fois qu'on veut obtenir un nombre aléatoire, on applique la formule :

```
z = f(z) = (a * z) mod m
```

où :

- mod est l'opérateur modulo (reste de la division entière) ;
- a, nommé multiplicateur, vaut 16807 ;
- m, nommé module, vaut :

```
2^31 - 1 = 2 174 483 647.
```

L'utilisation de ce GNA se fait au moyen de la fonction :

```
Randm() = f(z) / m
```

qui renvoie un réel (type float en C) compris entre 0 et 1 exclus.

LE JEU D'EXPLORATION SPATIALE A UN NOM !

Afin de prendre des exemples concrets, nous avons supposé que nous programmons un jeu d'exploration spatiale, où divers facteurs aléatoires rendent la vie du personnage du jeu très... hasardeuse. En effet, le perso découvre sur nombre d'astéroïdes des filons au rendement aléatoire pour aller dilapider son gain dans de sordides tripots !... Toute ressemblance avec une grande ville du Midi, etc.

Un petit concours était ouvert jusqu'à la fin du mois de novembre pour trouver un nom sonnant bien pour ce jeu. Merci à tous ceux qui y ont participé. L'heureux gagnant est Bestofmegamix, qui, malgré un pseudo très anglicisant, a trouvé l'élégant nom "Erz", ce qui signifie "minerai" en allemand et nous change des niaiseries anglo-saxonnes que l'on voit sur tous les "ticheurtes" des touristes.

DISTRIBUTIONS DISCRETES

"Distributions discrètes", quesaco ? N'allez pas vous imaginer des revendeurs louches qui fourguent en douce de l'anisette frelatée aux passants. Simplement, on peut distinguer deux types de variables aléatoires :

- celles qui prennent une valeur continue, exprimée par un réel, où toutes les valeurs dans un certain intervalle sont possibles. Un exemple : dans notre jeu Erz, quand le perso découvre un filon sur un astéroïde, le tonnage de métal précieux extractible est un réel à fixer entre deux limites correspondant à un gisement pauvre et à un gisement riche. Ces variables aléatoires sont dites distribuées (c'est-à-dire réparties) continuellement, ou à distribution continue ;
- celles qui prennent une valeur parmi un ensemble fini (ou discret) de valeurs possibles, et qui sont dites à distributions discrètes. Exemple : si le perso d'Erz entre dans un tripot et joue aux dés,chaque tirage de dé produit une variable qui peut prendre les valeurs de 1 à 6 (d'où 6 valeurs possibles seulement).

La distribution discrète que nous avons déjà rencontrée correspond précisément au cas d'un tirage de dé honnête (où toutes les faces du dé ont la même probabilité de sortir) : il s'agit de la distribution équiprobable. Le volet précédent propose le listing de la fonction Equiprobable(a, b) qui retourne un entier aléatoire entre a et b. Ainsi, pour simuler un tirage de dé à 6 faces (je les vois venir, les monstres qui vont me sortir un dé à 20 faces !), il suffit d'appeler la fonction **Equiprobable(1, 6).**

Au fait, et que se passerait-il si on lançait non plus seulement un seul



dé à la fois, mais deux (étant entendu que le résultat du tirage est alors la somme des points obtenus sur les deux dés) ? A l'évidence, le nombre minimal est alors 1 + 1 = 2, et le nombre maxi est 6 + 6 = 12. Mais examinons la figure 1. Elle représente un tableau de 6 x 6 cases où sont représentés tous les cas possibles pour le résultat des deux dés. Dans chaque case du tableau figure la somme des deux dés.
Comment ? Que dites-vous ? Pas très intellectuel jusque-là ? Le pastis ne contient visiblement pas de phosphore, dites-vous ? Je l'admets, mais voyez donc la figure 2 : c'est un tableau où l'on a fait correspondre chaque résultat possible, de 2 à 12 (première ligne) avec le nombre de fois où ce résultat apparaît dans la figure 1 (seconde ligne). Bien sûr, le total de ces nombres de fois fait 36 (si, si, vérifiez). On note que dans la figure 1, certains résultats apparaissent bien plus souvent que d'autres (le 7 apparaît 6 fois alors que le 2 ou le 12 n'y sont qu'une seule fois). Cela signifie que si vous misez sur le 7, votre chance de gagner est plus élevée que pour tout autre résultat. Autrement dit, sa probabilité est plus élevée. Mais au fait, comment calcule-t-on la probabilité d'obtenir le 7 dans ce cas ? Simple : il suffit d'appliquer la règle d'or des distributions discrètes, qui est :

Probabilité de X = (nombre de fois où apparaît X) divisé par (nombre de cas total).

Pour appliquer cette règle, encore faut-il avoir procédé comme ci-dessus, c'est-à-dire avoir dénombré tous les cas possibles. Ici, le travail est fait, et le nombre de cas total, nous l'avons vu, est 6 * 6 = 36. Donc, la probabilité d'obtenir un 7 est 6 / 36, soit 1/6, soit encore 0,1666... ou 16,7 % en arrondissant. De même, on calcule les probabilités des autres résultats. Exprimées en pourcentages, elles sont regroupées dans la 3e ligne de la figure 2.

Et, si l'on se reporte au volet précédent, on s'aperçoit que ce genre de tableau porte un nom : c'est la fonction de distribution de probabilité

```
FIGURE 1 : Somme de deux dés
Dé 1 :
     ,  ,  1, 2, 3, 4, 5, 6
     , 1,  2, 3, 4, 5, 6, 7
     , 2,  3, 4, 5, 6, 7, 8

Dé 2 :
     , 3, 4, 5, 6,  7,   8,  9
     , 4, 5, 6, 7,  8,   9, 10
     , 5, 6, 7, 8,  9,  10, 11
     , 6, 7, 8, 9, 10,  11, 12
```

(FDP) de la variable aléatoire ayant pour valeur le résultat du lancer de nos deux dés. Il est évident que proposer dans ces conditions une partie de dés où tous les résultats rapportent la même mise est impossible. Les gains doivent être proportionnels au risque de chaque mise. Et les risques, ce sont les probabilités des résultats, qu'on vient de calculer.

Et comment programme-t-on un jet de deux dés ? Facile : si r est une variable entière, il suffit de calculer :

```
r = Equiprobable(1, 6) + Equiprobable(1, 6)
```

r contiendra la somme des points obtenus par deux dés.

Attention : un lecteur peu attentif pourrait croire que l'on peut simplifier le calcul ci-dessus et faire :

```
r = 2 * Equiprobable(1,6)
```

Que nenni, messires ! Car la fonction Equiprobable, faisant appel à notre GNA Randm(), retourne à chaque fois un entier différent entre 1 et 6. Par conséquent, pour simuler le lancer de deux dés, il faut bien appeler Equiprobable(1, 6) deux fois. Si on se contente de l'expression 2 * Equiprobable(1,6), on obtient bien une valeur entre 2 et 12, mais gare : seules sont alors possibles les valeurs paires 2, 4, 6,..., 12, et de plus, ces valeurs sont équiprobables par définition de la fonction Equiprobable(), et la FDP n'est plus du tout celle de la figure 2.

LANCER DE DÉS MULTIPLES

Le jeu se complique

On peut se poser la question de savoir ce qui se passe pour un plus grand nombre de dés N. En effet, quand le perso d'Erz entre dans ce tripot, il est tentant de proposer au joueur un jeu de hasard inédit. Et miser sur le résultat du lancer de dix dés, par exemple, voilà qui n'est pas commun. Mais est-ce intéressant ?
Pour le savoir, il nous faut examiner la FDP de la variable aléatoire X, égale à la somme des points obtenus en lançant N dés (toujours à 6 faces), histoire de voir à quoi ressembleront les résultats obtenus. Pour tenter de découvrir cette FDP, deux types de méthodes s'offrent à nous : la méthode de la moyenne, qui est expérimentale, et la méthode du dénombrement, moins marrante mais plus efficace car donnant des résultats exacts. Comme les programmeurs sont en général des êtres imaginatifs dotés d'un fond rationnel, il nous semble logique de commencer par la première.

La méthode de la moyenne

Commençons par poser le problème. Nous voulons une approximation de la FDP de notre variable X ci-dessus, définie par :

```
X = somme des points obtenus en lançant N dés
```

Notons déjà qu'avec N dés, les résultats possibles (c'est-à-dire les valeurs x de X) sont compris entre N et 6 * N inclus. Mais quelle est la probabilité de chaque résultat ? Pour savoir cela, on se propose d'utiliser l'algorithme suivant :
- pour chacun des résultats possibles, définir un compteur initialement nul, nommé c[x] ;
- faire un grand nombre NT de tirages, et comptabiliser soigneusement le nombre de fois où l'on obtient chaque résultat : si on obtient x lors du tirage, on incrémente c[x] ;
- après les NT tirages, "au pif", dire que si un résultat x a été obtenu c[x] fois (c[x] étant la valeur du compteur correspondant à x après les NT tirages), la probabilité p du résultat x est :

```
p = Pr{X = x} = c[x] / NT
```

```
FIGURE 2 : Probabilités des résultats de la somme de deux dés
Somme     : 2   , 3   , 4   ,  5   ,  6   ,  7   ,  8   ,  9   ,10  ,11  , 12
Nb.fois   : 1   , 2   , 3   ,  4   ,  5   ,  6   ,  5   ,  4   , 3  , 2  , 1
Proba(%)  : 2.78, 5.56, 8.33, 11.11, 13.89, 16.67, 13.89, 11.11, 8.33, 5.56, 2.78
```

Deux observations qui prouvent que Tonton Billy T. a encore la tête claire bien que les gendarmes sur la route lui aient trouvé un peu de sang dans son alcool :

- comme il y a NT tirage et qu'à chaque tirage, on incrémente l'un des compteurs, la somme des compteurs fait NT. Donc la somme des probabilités Pr{X = x} pour toutes les valeurs possibles de x (de x = N à x = 6 * N) est :

```
Pr{X=N} + Pr{X=N+1} +...+ Pr{X = 6*N}
 =(c[N]/NT)+(c[N+1]/NT)+...+(c[6*N]/NT)
 =(c[N]+c[N+1]+...+c[6*N])/NT
 = NT / NT
 = 1.
```

Donc la somme des probabilités de tous les résultats possibles est bien 1, ce qui est indispensable pour une loi de probabilité bien née ;

- le "pif" dont il est question plus haut consiste en fait à faire une moyenne des résultats sur un grand nombre d'essais. Intuitivement (vous n'intuitivez pas, vous ? Voilà ce que c'est d'être toujours à jeun !), on conçoit que les probabilités obtenues ainsi tendent vers les valeurs exactes lorsqu'on augmente le nombre d'essais : les fluctuations statistiques sont "noyées dans la masse", telle la modeste bouteille dans ma consommation mensuelle. Cette règle pifométrique est parfaitement exacte, bien que compliquée à prouver formellement, et porte le doux nom de **loi des grands nombres.**

Notons que cette méthode est aussi connue sous le nom de **méthode de Monte-Carlo**, car elle s'en remet aux statistiques (la loi des grands nombres) pour établir un résultat, et consiste en fait à faire des expériences de tirages aléatoires, tout comme les croupiers monégasques (mais eux le font dans des buts nettement moins bénévoles). Dans certains cas, cette méthode est la seule valable, car si les événements à simuler deviennent très complexes (comme certains problèmes de physique nucléaire ou d'astrophysique), la science est incapable de donner une équation permettant de calculer d'avance les probabilités des événements. Ce n'est pas le cas pour notre tirage de dés, et en plus de la méthode des dénombrements, on peut calculer les valeurs exactes des probabilités des résultats.

Le listing 1 (fichier "meth_moy.c") donne un exemple de programme qui implémente l'algorithme décrit ci-dessous.

Analyse du programme meth_moy.c

Le nombre de tirages est la variable nt, définie comme un long, et saisie par scanf() (attention, vérifiez que votre compilateur accepte le format %ld pour saisir un long avec scanf(). Certains exigent le format %d). De même, le nombre de dés n est saisi. Les compteurs sont stockés dans le tableau long c[] dimensionné à NBCOMPTEURS, lui-même défini à 5001. Cela vous permet de simuler jusqu'à n = 1000 dés (défini dans la constante NBMAXDES), ce qui est largement suffisant. Pourquoi 5001 ? Parce que les valeurs possibles x de X vont de n à 6 * n, soit (6 * n - n + 1) = 5 * n + 1 valeurs différentes, ou encore 5001 valeurs différentes pour n = 1000. Ce dimensionnement à l'avance du tableau c[] permet d'éviter de se plonger dans des malloc() : restons simple. La boucle while permet d'entrer un nombre de dés par tirage entre 2 et NBMAXDES.

Les compteurs sont dans un tableau, donc le premier est d'indice 0. Or le premier compteur correspond à X = n. Donc l'indice du compteur s'obtient en soustrayant n à la valeur du résultat de la somme des n dés. Une boucle simple permet de remettre à zéro les compteurs c [ ].

Puis on effectue nt tirages. Pour chaque tirage, on additionne dans res (prélablement remis à zéro) la valeur de n appels à la fonction Equiprobable(). La valeur de **res** sert ensuite d'index pour savoir quel compteur incrémenter : si l'on obtient 5, on incrémente le compteur numéro 5, pour "se rappeler" que le 5 est sorti à ce tirage.

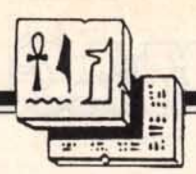
LISTING 1

```
/*==== fichier meth_moy.c ====*/
#include <stdio.h>
#define NBMAXDES 1000                     /* Nombre maxi de dés */
#define NBCOMPTEURS (5 * NBMAXDES + 1)  /* Nombre compteurs */

extern float Randm(long);                     /* Voir St Mag 45 */
exterm int Equiprobable(int, int);

main()
{
  long c[NBCOMPTEURS];              /* Compteurs de résultats */
  long nt;                         /* Nombre total de tirages */
  int n;                 /* Nombre de dés jetés simultanément */
  int i;
  int res;                      /* Résultat d'un jet de n dés */
  long l;
  int ok = 0;

  Randm(-1L);                /* Init GNA avec horloge système */
  printf("Entrez le nombre total de tirages:");
  scanf("%ld", &nt);     /* %D ou %ld pour le Mark Williams C */
  while (ok == 0)
  {
  printf("Entrez le nombre de dés (entre 2 et %d):", NBMAXDES);
  scanf("%d", &n);
  if ((d >= 2) && (d <= NBMAXDES))
    ok = 1;
  else
    printf("Mauvaise valeur !");
  }
```



```
/* Mise à zéro des compteurs */
 for (i = 0; i < 5 * n + 1; i++)
   c[i] = 0L;

 /* Boucle de nt tirages */
 for (l = 0L; l < nt; l++)
 {
   res = 0;
/* Met dans res la somme de n dés */
   for (i = 0; i < n; i++)
     res += Equiprobable(1, 6);

/* Incrémente le compteur corresp.*/
   c[res - n] ++;
 }

 /* Affichage des résultats */
 printf("Résultat des %ld tirages
        obtenus (entre %d et %d)\n\n",
        nt, n, 6 * n);

 for (i = 0;  i <= 5 * n; i++)
   printf("%d est sorti %d fois
sur %ld, soit %9.4f %%\n",i + n,
c[i], nt, (float)c[i] * 100.0 / nt);
}

/*==== fin fichier meth_moy.c ====*/
```

Enfin, on affiche les résultats, sous la forme "5 est sorti 124 fois sur 2000, soit 6.12 %", pour toutes les valeurs de x entre n et 6*n. Le pourcentage est obtenu en divisant la valeur du compteur par le nombre de tirages, et, comme on veut un pourcentage, on multiplie par 100 (exprimé comme un float pour obtenir une valeur du même métal).

La méthode du dénombrement

La méthode du dénombrement, contrairement à la précédente, n'est pas basée sur l'expérimentation statistique, mais sur un dénombrement systématique des résultats possibles. Pour chaque résultat, on incrémente un compteur, comme précédemment. La difficulté consiste à recenser tous les cas possibles. Une méthode évidente consiste à faire une boucle pour chaque dé, et à imbriquer les boucles. Le problème est que pour 8 dés, par exemple, on se retrouve au total avec 1.679.616 itérations. Et ça rame ! Il existe des méthodes plus complexes mais plus performantes, que nous n'aborderons pas ici, vive la force brute ! En réalité, l'objet ici n'est pas de découvrir des algorithmes de dénombrements mais de simuler des dénombrements. Il ne faut pas que la méthode complexifie trop cette série. Le listing 2 ci-contre (fichier "meth_den.c") contient une implémentation de la méthode des dénombrements.

Ici aussi, comme dans meth_moy.c, on utilise 5 * n + 1 compteurs dans le tableau c[ ]. Par contre, il faut implémenter un nombre n variable de dés, et faire en sorte que tous ces dés prennent successivement les valeurs de 1 à 6. Autrement dit, on veut construire un tableau à n chiffres qui est initialisé avec tous ses chiffres à 1 et compte de 1 en 1, chaque chiffre repassant à 1 après le 6 et envoyant une retenue au chiffre suivant. Ainsi, les n dés valent :

```
111...1, 211...1, 311...1, 411...1,
511...1, 611...1, 121...1, 221...1,
321...1, etc.
```

Ainsi, on est sûr que tous les cas possibles seront bien dénombrés. Les n dés sont stockés dans le tableau de [ ] dimensionné à NBMAXDE comme précédemment. Cependant, ici, NBMAXDE vaut 12 seulement, car le nombre de cas total est 6^n, qui vaut déjà plus de 2 milliards quand n = 12, ce qui suffit à occuper un gros système pendant un temps considérable (a fortiori un ST).

On procède comme précédemment pour lire le nombre de dés jetés simultanément. Ensuite, une autre boucle calcule le nombre de cas possibles, qui est 6 puissance n. La boucle permet d'éviter un calcul en passant par les logarithmes et exponentielles. Une boucle pour mettre à zéro les compteurs, une pour mettre à un les dés, et nous voilà parés pour la boucle de dénombrement (le while(r = 0)). La variable r contient la retenue du tableau des dés. A chaque dénombrement, on fait la somme des valeurs des dés contenues dans de[0] jusqu'à de[n - 1], et on utilise cette somme pour savoir quel compteur incrémenter. Puis l'on incrémente le premier chiffre du tableau des dés, et, s'il valait déjà 6, ce dé repasse à 1, et la retenue passe à 1. On passe au chiffre suivant et ainsi de suite. Si l'un des dés vaut moins de 6 (ce qui est le cas avant la dernière itération), ce dé sera incrémenté, la retenue passera à 0 et le break provoquera une sortie de la boucle for. Si tous les dés sont à 6, la retenue reste à 1, et la condition du while devient fausse, d'où fin de la boucle while.

DISTRIBUTION CONTINUE

Jusqu'ici, les événements aléatoires que nous avons simulés étaient toujours discrets. Mais il arrive que cette approche ne suffise pas. Reprenons l'exemple d'Erz : le vaisseau du perso navigue dans le vide interstellaire et croise un vaisseau pirate qui cherche à engager le combat ! Deux paramètres importants : le cap et la vitesse du pirate. Le cap s'exprime sous la forme de deux angles (holà, géométrie en 3D, laissons cela), et la vitesse sous forme d'un nombre. Pour la vraisemblance, la vitesse, par exemple, devrait être quelconque. Elle doit être prise au hasard entre deux valeurs limites, et être un nombre réel. Autrement dit, n'importe quel réel entre deux bornes a la même probabilité d'être choisi pour être la variable aléatoire vitesse ; cela s'appelle une **distribution uniforme**, et c'est l'équivalent pour le domaine continu de la distribution équiprobable dans le domaine discret que nous avons déjà vu.

Et comment engendrer une variable uniforme ? Il suffit d'observer que Randm() est déjà une variable distribuée uniformément entre 0 et 1 exclus ! Donc, pour engendrer une variable uniformément distribuée entre a et b exclus, il suffit d'employer la fonction Uniforme(a, b) définie comme suit :

```
float Uniforme(a, b)
float a, b;
{
  extern float Randm();

  return((b - a) * Randm() + a);
}
```

Halte au feu, me crierez-vous ! Randm() n'est pas VRAIMENT une distribution réelle (pouvant donner un nombre infini de valeurs), car comme le générateur de Lehmer ne prend que 2^31 - 1 valeurs distinctes, Randm() n'en prend pas plus, et en fait, il y a un nombre très

LISTING 2

```
/*=== fichier meth_den.c ===*/
#include <stdio.h>
#define NBMAXDES 12                    /* Nombre maxi de dés */
#define NBCOMPTEURS (5 * NBMAXDES + 1)  /* Nombre compteurs */

extern float Randm(long);                      /* Voir St Mag 45 */
exterm int Equiprobable(int, int);

main()
{
  long c[NBCOMPTEURS];              /* Compteurs de résultats */
  int n;                  /* Nombre de dés jetés simultanément */
  int i;
  int de[NBMAXDES];                        /* Contient les n dés */
  int res;                         /* Résultat d'un jet de n dés */
  float tot;      /* Contient 6^n = nombre de cas possibles */
  int ok = 0;
  int r;                  /* Retenue pour incrémenter les dés */

  Randm(-1L);                   /* Init GNA avec horloge système */

  while (ok == 0)
  {
  printf("Entrez le nombre de dés (entre 2 et %d):", NBMAXDES);
scanf("%d", &n);
  if ((d >= 2) && (d <= NBMAXDES))
    ok = 1;
  else
    printf("Mauvaise valeur !");
  }

  tot = 1.0;                            /* Calcul de tot = 6^n */
  for (i = 0; i < n; i ++)
    tot *= 6.0;

  for (i = 0; i < 5 * n + 1; i++)  /* Mise à zéro des compteurs */
    c[i] = 0L;

  for (i = 0; i < n; i++)           /* Mise a un des dés */
    de[i] = 1;

  r = 0;              /* Boucle de comptage  */
  while (r = 0)   /* Tant qu'on n'a pas tous les dés a 6 */
  {
    res = 0;
    for (i = 0; i < n; i++)  /* Met dans res la somme de n dés */
      res += de[i];

    c[res - n] ++;  /* Incrémente le compteur correspondant */

    /* "Incrémente" les dés en propageant une retenue */
    r = 1; /* retenue */
    for (i = 0; i < n; i++)
    {
      de[i] += r;
      if (de[i] > 6)                /* Si le dé valait 6... */
      {
        de[i] = 1;                  /* ...il passe à 1...   */
        r = 1;             /* ... et on propage la retenue. */
      }
      else                     /* Si non, sort de la boucle */
      {
        r = 0;
        break;
      }
    } /* fin for */
  }   /* fin while */

  /* Affichage des résultats */
  printf("Résultat des %f tirages obtenus (entre %d et %d)\n\n",
tot, n, 6 * n);

  for (i = 0;  i <= 5 * n; i++)
    printf("%d a été dénombré %d fois sur %f, soit %9.4f %%\n",
i + n, c[i], tot, (float)c[i] * 100.0 / tot);
}

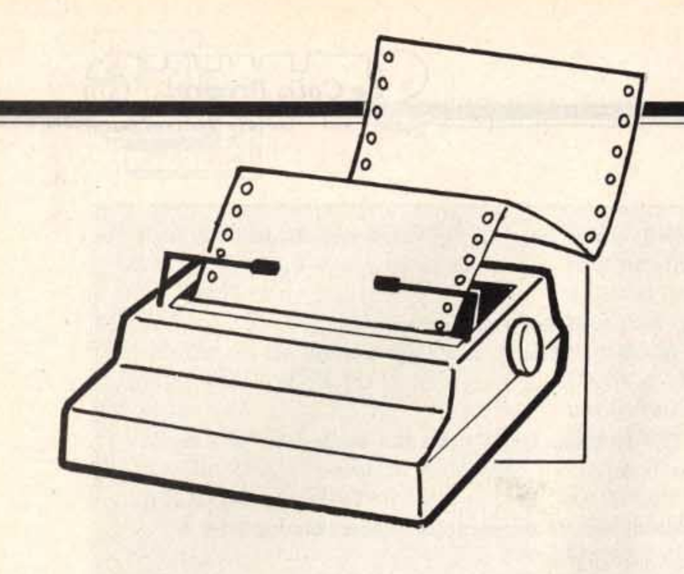
/*==== fin fichier meth_den.c ====*/
```

élevé mais fini de valeurs distinctes. Chipotages, vous répondrais-je : notez aussi que comme un nombre à virgule flottante est stocké avec un nombre fini de bits (32 ou 64 en général), il peut prendre lui aussi un nombre fini de valeurs. Donc, nous considérerons que la fonction Uniforme() remplit bien son rôle et donne un nombre infini de valeurs.

D'autre part, on observe que l'on est aussi limité par la précision du modèle que l'on emploie : à quoi bon afficher des vitesses en mètres/seconde avec 5 chiffres après la virgule, si le joueur ne peut régler la vitesse de son vaisseau qu'à 1 km/s près ? En fait, on verra que dans l'énorme majorité des cas, il suffit de trouver une subdivision suffisamment fine de la grandeur physique à simuler (ici, la vitesse) puis de l'utiliser exclusivement. Cela nous ramène alors à un nombre fini de valeurs possibles pour chaque événement, ce qui permet d'utiliser des distributions discrètes, et souvent de travailler presque uniquement avec des entiers (ce qui accélère les programmes).

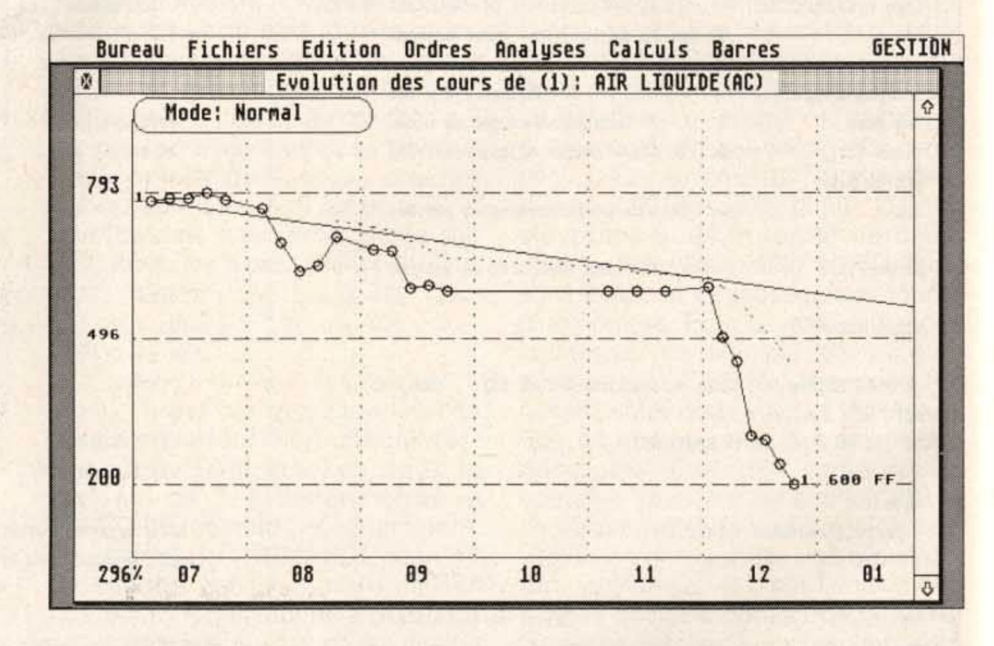
En guise de matière à réflexion jusqu'à ce que nous nous retrouvions dans le 5e volet de cette série, mes p'tits neveux, Tonton Billy T. vous propose de considérer une variable aléatoire X qui est obtenue par Uniforme(1, 10). Quelle est la probabilité de X = 3 ? Comme il y a une infinité de valeurs réelles possibles entre 1 et 10, et un seul réel valant 3, cette probabilité est égale à 1 divisé par l'infini, c'est-à-dire zéro, non ? Mais si c'est le cas pour X = 3, c'est aussi vrai pour tous les réels entre 1 et 10, qui ont tous la probabilité 0, non ? Donc, aucune valeur pour X n'est possible, puisqu'elles sont toutes de probabilité nulle, pas vrai ? Mais alors, pourquoi Uniforme(1, 10) s'obstine-t-elle à fournir des résultats, puisque aucun d'eux n'est possible ???!!! Cherchez l'erreur, et à la prochaine.

Billy T.



FORTUNE

La logithèque du ST comptait déjà bon nombre de logiciels de gestion de comptes et s'est récemment enrichie de logiciels d'analyse boursière. Fortune, logiciel de gestion boursière et des biens particuliers édité par Arobace, fait enfin la synthèse et vous permettra de gérer sur un même programme l'ensemble de votre... fortune.



Grâce à votre dur labeur quotidien, voire à l'héritage d'un oncle d'Amérique, vous disposez de quelques économies. Qu'elles se composent d'un livret A bien rempli, d'un appartement à Paris, d'une parure de famille ou d'un portefeuille de valeurs mobilières internationales, Fortune se propose de vous aider à les gérer au mieux.
Liquidités, titres (actions, obligations, bons de souscription, OC ou tout autre instrument financier), tel ou tel bien de votre patrimoine : jusqu'à 13 comptes peuvent ainsi être définis. Il suffit ensuite d'enregistrer sur ces comptes toutes les opérations que vous effectuerez (achats, ventes, versements, retraits...) : Fortune calculera immédiatement l'état de vos finances.

Là-dessus, on peut déjà regretter que le logiciel ne calcule pas automatiquement les intérêts de liquidités placées sur un compte rémunéré (livret A, Plan ou Compte d'épargne logement)... Vous devez donc périodiquement ajouter à la trésorerie le montant des intérêts perçus sur votre compte : cette lacune devrait néanmoins disparaître dans la prochaine version.

Fortune vous sera surtout d'une grande utilité pour la gestion de votre portefeuille de valeurs mobilières. Grâce au rapatriement automatique des cours du Minitel via les serveurs spécialisés (3615 VF, TRIBUNE, ou bien d'autres), ou à l'entrée directe de ceux-ci par le clavier, vous connaissez à tout moment votre potentiel de plus-value sur un titre ou sur l'ensemble de votre portefeuille.

Mais Fortune met également à votre disposition différents outils d'analyse, en vous permettant de visualiser les répartitions géographiques et sectorielles de vos placements, et surtout en créant des graphiques de cours sur 3 mois (soit 30 points au maximum), 6 mois (120 points) ou 18 mois (uniquement sur les cours extrêmes), avec une moyenne mobile. Sur un même schéma, il est possible de visualiser plusieurs graphiques ; une option très pratique, pour comparer par exemple l'évolution d'un bon par rapport à celle de son titre support, ou l'évolution d'un titre par rapport à un indice... De plus, ces graphiques sont exploitables à l'aide de la souris : en déplaçant celle-ci sur une courbe, vous voyez apparaître la valeur approchée correspondant à la position du pointeur. Les amateurs de "Charts" trouveront également leur compte en traçant des lignes de crête, d'épaule, et tout ce qui leur semblera utile. Point d'importance : le logiciel gère automatiquement les échelles.
D'autre part, Fortune possède une option de génération automatique des ordres, selon les cours limites d'achat ou de vente que vous vous êtes fixés. En cas d'opération financière (augmentation de capital, émission d'ABSA....), vous pourrez ajuster très simplement l'ensemble des cours via une option spécifique,

et même calculer la valeur d'un bon de souscription. Le logiciel enregistre évidemment toutes vos opérations : versements, retraits, achats, ventes...

Ceci vous permettra à tout moment de connaître votre trésorerie et votre performance sur un compte particulier ou sur l'ensemble. Les données sont sauvegardées au fur et à mesure de la saisie : cela vous assure une sécurité maximale. Même si vous êtes très riche, et ne possédez qu'un lecteur de disquettes (le cas existe-t-il encore ?), le temps de sauvegarde restera tout à fait correct. Time is Money ! Une option "Journal" passe en revue l'ensemble des opérations que vous avez effectuées, et l'option "Bilan" évalue l'état de votre patrimoine.

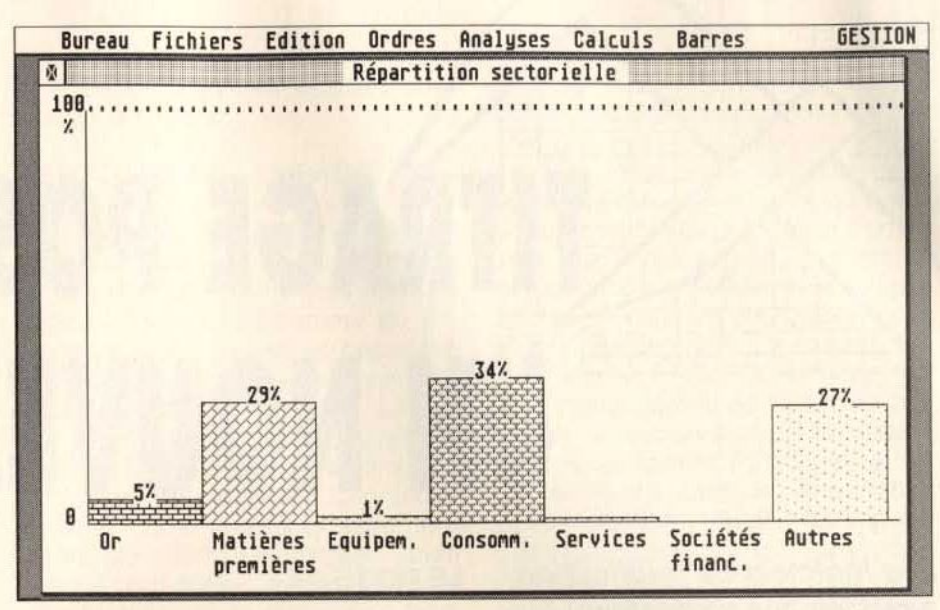
Mais attention, aussi pratique soit-il, Fortune n'est qu'un outil d'aide à la gestion ; il ne prétend pas prendre de décision à votre place et ne réalise pas de prévision... Ça, c'est le rôle des analystes financiers, et aucun logiciel n'en est (encore) capable.

LA SAISIE DES DONNÉES

La saisie manuelle se fait conjointement à la souris et au clavier, grâce à des tableaux bien fournis et bien présentés. Cependant, il est regrettable que les mêmes masques de saisie servent à l'entrée de biens aussi différents qu'une action ou un bien immobilier. De plus, il n'est pas rare qu'après avoir passé quelques minutes à remplir un tableau, vous ayez à le ressaisir en entier, alors qu'une seule des valeurs entrées était erronée. L'économie de programmation et de mémoire se fait un peu trop ressentir chez l'utilisateur, qui pourra reprocher au logiciel une ergonomie imparfaite.

LE MANUEL

Il se présente sous la forme d'un classeur (pratique), ce qui autorise des mises à jour documentées. Sans se perdre dans des détails superflus, les 70 pages présentent les nombreuses options de Fortune. On regrettera cependant (bis) l'absence d'un exemple complet permettant la prise en main du logiciel. Un index et un lexique auraient également été les bienvenus.



LE CODE (ou l'aventure de l'amant démasqué)

Lors du premier lancement du programme, Fortune vous demande d'entrer votre code hyperconfidentiel, qui devrait protéger l'accès à vos comptes. Enfin, en principe. Car si votre femme a la curiosité de double-cliquer sur un fichier dont nous tairons le nom, elle verra apparaître en clair (et sans décodeur) votre code secret en haut de l'écran. Moralité : ne revendez pas vos Saint-Gobain pour acheter un vison à votre maîtresse !

CONCLUSION

Fortune s'avère être un logiciel de gestion de patrimoine des particuliers plutôt complet sur ST, notamment de par ses outils d'analyse et de suivi boursiers très performants.

Les quelques imperfections que nous avons relevées devraient, à n'en pas douter, être éliminées dans la prochaine version, attendue pour le printemps 91 ; la mise à jour vous coûtera alors une centaine de francs.

Michèle Wolff

CONFIGURATION REQUISE

Atari ST avec 1 Mega de RAM
Lecteur de disquettes 720 ko
Écran monochrome 640 x 400
Imprimante (compatible avec Hard Copy par Alternate+Help) et Minitel facultatifs.

Prix public TTC inférieur à 1500 F

Bureau Fichiers Edition Opérations Comptabilité Barres SUIVI

Bilan du 01/01/90 au 16/12/90

	Actif	Passif
Début d'exercice		0
Trésorerie		2.271.168
Portefeuille	1.446.420	
Résultat	824.747	
	621.673	621.673
Analyse:	Montant	Rendt
- Coupons	0	ns. %
- (Apports)	2.100	
- M-Value	-826.847	100 %
Perte	-826.847	100 %

(d'après options de comptabilistion)
(opérations à terme incluses)



TITRAGE POSTSCRIPT : UN NOUVEAU VENU

Les logiciels de publication assistée par ordinateur sont de plus en plus souvent associés à des logiciels de création de logos et de titrage, comme Outline-Art qui s'est imposé comme complément de Calamus. Publishing Master se voit doté, avec Lasergraph Pro, d'un assistant qui lui faisait cruellement défaut.

UN LOGICIEL POSTSCRIPT

Il n'existe pas pour le moment de logiciel de dessin PostScript réellement adapté au titrage et à la création de dessins, logos, s'adressant aux imprimantes PostScript. Laser-Graph Pro, édité par la société "Editia", vient combler cette lacune, et permet la création de dessins variés, offrant des possibilités de dégradés et de contours variables.

LES OUTILS DE BASE

Les outils de base sont classiques. Ils comportent droite, rayon, polyligne, rectangle (bords droits ou arrondis), cercle, ellipse, texte, auxquels s'ajoute la possibilité de créer les fameuses courbes de Bézier. Il est possible d'affecter une grille paramétrable, magnétique ou non, de manière à guider le tracé.

LE TEXTE

La création d'un texte se fait en deux temps : on effectue la saisie dans une boîte spéciale, en mode droit ou cercle (le texte suivra alors un cercle). Les paramètres du texte, comme l'épaisseur de ligne (ou le motif de fond) doivent être réglés avant de le positionner, mais celui-ci peut être changé, caractère par caractère, une fois le texte placé.
On place le texte sur le dessin en cliquant à l'endroit désiré, en indiquant les points de début et de fin. Il est possible de régler la taille, l'inclinaison des caractères, l'interlettrage. Lorsque le texte est placé, il n'est pas possible de le modifier, puisqu'il n'existe plus qu'à l'état vectoriel, c'est-à-dire de dessin, et il faut alors l'effacer, et le replacer, après avoir corrigé les éventuelles fautes. Le texte peut aussi suivre une courbe de Bézier, de manière à produire des ondulations. Le texte est alors converti en lignes (courbes de Bézier), et certains caractères sont alors formés de plusieurs objets non groupés, rendant difficile une manipulation ultérieure.

LES OUTILS DE POSITIONNEMENT

Un mode "voir" permet d'afficher le dessin avec tous ses attributs, alors que le mode de travail n'affiche que les contours pour la plupart des objets.

Il est possible d'effectuer une rotation des objets au degré près, et d'en aligner plusieurs ensemble. On peut placer les objets devant ou derrière, et grouper ou séparer les éléments. Les objets peuvent être déplacés



après sélection, à l'aide de la souris, au prix de manipulations plutôt étranges. En effet, l'effet produit n'est pas toujours celui escompté (rarement même, pour être franc), ce qui semble dû à une gestion de la souris assez particulière, comme en témoigne l'incessant clignotement du pointeur. Même en respectant scrupuleusement la documentation, il est difficile d'obtenir le résultat escompté dès le premier essai ; renseignements pris, le développeur travaille d'ailleurs à l'amélioration de ces fonctions. Il est possible de créer une surface à partir d'éléments dessinés, de manière à lui affecter un motif de fond, à condition de donner un ensemble d'objets formant un contour fermé.

Il est possible d'accéder, via la touche F2, à un écran de travail supplémentaire, afin de récupérer des fragments d'un autre document, ou de l'utiliser comme table de travail. La plupart des fonctions y sont accessibles, y compris le presse-papier, mais il n'est pas possible d'imprimer un document à partir de ce deuxième écran.

LE PRESSE-PAPIER

Il est possible de couper /copier /coller des objets, par l'intermédiaire des menus, ou en appelant des raccourcis clavier. Il est possible de plus de choisir une longueur de translation, et un angle de rotation, de manière à effectuer ces deux opérations directement lors de la duplication.

LES MOTIFS

Il est possible de sélectionner les motifs de lignes et contours, et les fonds de remplissage. On choisit un motif, un niveau de gris, et l'angle de trame (pour les lignes). Un mode transparent est possible, qui devrait permettre de venir en surimpression, sans masquer un objet qui se trouverait derrière. En réalité, on obtient seulement le contour de l'objet, et le motif de fond n'est plus pris en compte. Il est possible d'effectuer des dégradés sur les rectangles et les ellipses, dont le résultat à l'écran est peu convaincant, obligeant à imprimer la page pour vérifier le résultat.

LES POLICES DE CARACTERE

Il est possible d'employer simultanément huit polices, au format PostScript ou Publishing Master. Ces polices, fournies au nombre de 24, peuvent être visualisées caractère par caractère, mais ne peuvent être modifiées directement, ce qui est fort regrettable ! Le seul moyen de déformer un caractère semble être de donner un coefficient d'agrandissement différent dans le sens horizontal du coefficient vertical : bien maigre consolation. Ajoutons encore un bémol : dans la version que nous avons testée, l'implémentation de la police Dingbats nous est apparue des plus fantaisistes, mais il y a plus grave : aucune police ne nous a semblé pouvoir générer d'accents aigus, graves ou circonflexes.

L'IMPORTATION

Lasergraph Pro permet d'importer des images Bitmap, afin de les vectoriser manuellement. Les formats reconnus sont DEGAS et STAD. Elles sont alors placées sur un calque non imprimable ; il est possible de reprendre ces images en redessinant vectoriellement par dessus : quelle complication, alors que des logiciels de vectorisation d'images existent, même sur ST...

Il est de même possible d'importer des dessins créés par le logiciel lui-même, afin de fusionner des logos. Le format EPSF ("encapsulé PostScript"), qui permet l'importation directe de fichiers PostScript dans un document, n'est pas encore supporté, mais il semble qu'une prochaine version de Lasergraph Pro le fera, puisque l'option est prévue, mais non encore accessible.

Les dessins créés peuvent être exportés au format EPSF, pour être réutilisés dans un programme tel que Publishing Master, sauvés au format Lasergraph, ou encore en Bitmap (format Degas) : ce dernier format présente un grand intérêt. En effet, la plupart des logiciels comme Publishing Master permettent d'importer des fichiers au format PostScript, mais se contentent d'afficher une boîte vide à l'écran, interdisant de superposer de nouveaux objets à un dessin importé ou encore de le visualiser. Le format Encapsulé PostScript prévoit pourtant un fichier mixte (PostScript et image bitmap), mais n'est pas utilisé par la plupart des logiciels. La sauvegarde des images LASERGRAPH au format Degas permet de pallier ce problème. On sauve alors le dessin produit au format PostScript et au format Degas. Lors de l'importation dans Publishing Master, on importe l'image Degas, on lui superpose l'image PostScript : on peut alors voir l'image importée, et ajouter du texte ou d'autres objets. Avant d'imprimer le document, on enlève alors l'image Degas de manière à ne pas superposer une image Bitmap (de mauvaise qualité) à l'image PostScript qui, elle, est de bonne voire excellente qualité. Le nombre de manipulations est conséquent, mais cette possibilité est très utile.

CONCLUSION

Lasergraph Pro est un logiciel qui promet beaucoup, mais dont la réalisation n'est pas encore à la hauteur des espérances. Il permet néanmoins d'obtenir des images vectorielles Postscript de très bonne qualité, au prix de manipulations hélas souvent fastidieuses. Ce logiciel étant encore en phase de tests, des nouvelles versions devraient rapidement voir le jour. Signalons par ailleurs que le temps de publier cet article, une version nouvelle nous est parvenue, dont le présent article tient compte. La société Editia, qui réalise ce logiciel, travaille de plus sur des convertisseurs de formats permettant d'obtenir des fichiers au format Lasergraph à partir de fichiers PostScript, et donc de modifier dans Lasergraph les images EPSF créées avec d'autres logiciels. La seule difficulté de réalisation de tels convertisseurs réside dans la grande diversité des formats de ces fichiers EPSF.

Denis Delbecq

MINIX 1.5 : L'AGE DE RAISON

Il y a un peu plus d'un an, ST Mag présentait la version 1.1 d'un curieux animal qui prétendait transformer n'importe quel ST en un système fonctionnant sous UNIX v.7. MINIX ST 1.1 marchait, il marchait même très bien, mais un certain manque de finition dans la réalisation et une documentation réduite rendait son utilisation difficile pour ceux n'ayant pas déjà une certaine familiarité avec le monde UNIX. Aujourd'hui, le professeur Tanenbaum et ses associés nous proposent une version remaniée de MINIX, avec, comme nous allons le voir, pas mal de nouveautés.

BIENVENUE AU CLUB !

Les premières versions de MINIX avaient été accueillies avec enthousiasme dans le monde des utilisateurs d'UNIX. Très rapidement, informations, corrections de bugs et nouvelles commandes se sont mises à circuler sur les réseaux internationaux. La nouvelle version de MINIX a grandement profité de cet échange d'idées, et doit certainement représenter un exemple unique d'interaction entre les auteurs d'un logiciel pour micro et ses utilisateurs.

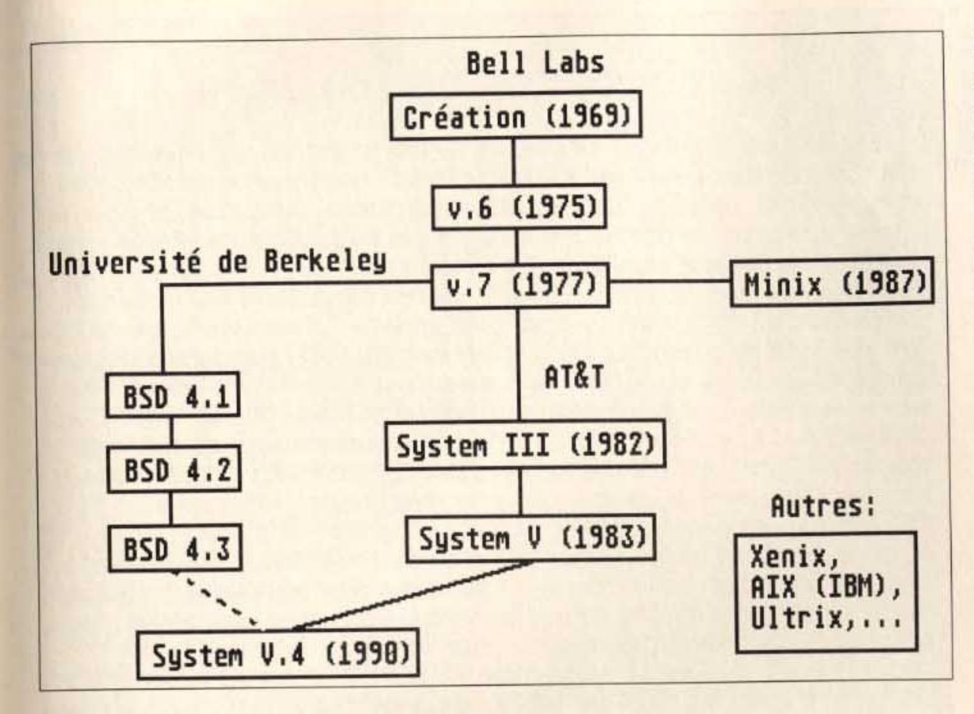
En arrivant à maturité, MINIX-ST a pris du poids : 2 disquettes simple face et 8 double face, un manuel (en anglais) d'environ 700 pages, contre 7 disquettes simple face et un fascicule de 60 pages pour la version précédente ! Le tout dans une boîte imposante, dont les dimensions font penser aux emballages monstrueux qu'affectionnent Borland et Microsoft. Le manuel, sur lequel nous reviendrons par la suite, possède une particularité : les chapitres ayant trait à l'installation mis à part, son contenu est indépendant de la machine hôte. En effet, MINIX 1.5 sort simultanément en version PC 3.5" et 5"1/4, Atari, Amiga, et Macintosh : seules les disquettes changent ! Si la version pour Amiga tient encore de l'anecdote (elle ne fonctionne pas avec un disque dur !), celle pour Macintosh semble bien intéressante : MacMINIX tourne dans une fenêtre, et serait compatible MultiFinder.

CONFIGURATION MATÉRIELLE

En principe, MINIX ST 1.5 est toujours capable de fonctionner avec seulement 512 ko de mémoire et un lecteur de disquettes, à condition que ce dernier soit un lecteur double face. Il est encore possible de disséminer le contenu des disquettes d'origine sur des 360 ko, mais l'utilisation de MINIX sera rendue très laborieuse. L'usage d'un 1040 avec un lecteur double face constitue donc un minimum, et la présence d'un disque dur est souhaitable.

Tel qu'il est fourni, MINIX est configuré pour l'utilisation sur un clavier américain. Un petit programme TOS, FIXKEYS.PRG, assure l'adaptation de MINIX aux spécificités de chaque clavier. Cette manipulation se fait en écrivant directement dans le fichier exécutable de MINIX : c'est une opération qui n'est pas sans danger (gare aux originaux !), et qui, de plus, semble avoir quelques inconvénients (voir encadré). Néanmoins, il faut faire cette manipulation au moins une fois, avant de pouvoir lancer le système.

Si vous avez la chance de posséder un disque dur, utiliser MINIX devient un vrai régal. MINIX sait gérer jusqu'à 4 partitions de 32 mégas sur un disque, et reconnaît la plupart des modèles du marché, y compris ceux au standard SCSI. Vous serez d'ailleurs rapidement fixé sur votre sort : après avoir lancé MINIX à partir d'une disquette, une commande



spéciale vous permet de voir si le système et le contrôleur de votre disque dur parlent la même langue. MINIX a pour l'instant été testé sur un Megafile 30, un disque amovible de 44 Mo et un Quantum SCSI de 80 Mo. Notez bien que l'installation de MINIX sur une partition de votre disque dur rend celle-ci totalement inutilisable pour le TOS : les deux systèmes de gestion de fichiers n'ont rien en commun. D'ailleurs, l'ouverture d'une fenêtre sur une partition ou sur une disquette MINIX donnera des résultats visuellement très spectaculaires ! Il est conseillé de faire une sauvegarde complète de votre disque dur avant de commencer toute procédure d'installation.

INSTALLATION

Comme la majorité des éditeurs de logiciels, Prentice Hall a la fâcheuse habitude de fournir les disquettes avec le volet de protection en position d'écriture. Prenez bien soin de protéger toutes les disquettes originales et de les recopier avant toute autre opération.

Le manuel est très précis sur toutes les étapes de configuration de MINIX : prise en compte de la mémoire vive disponible, utilisation d'une éventuelle horloge interne, installation sur un disque dur. Dans ce dernier cas, un fichier de commandes shell est fourni pour automatiser la création des répertoires et la recopie du contenu des disquettes. Pour installer tout le système, y compris les sources décompressés, il faut compter un minimum de 10 Mo. Si l'on veut se passer des sources, 6-7 Mo suffiront amplement. Installer MINIX est une opération qui peut prendre du temps (surtout si l'on veut tout mettre sur le disque dur), et qui demande de l'attention. Rien de bien difficile somme toute, mais la lecture attentive du manuel est indispensable.

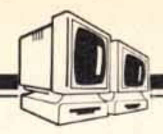
QUOI DE NEUF, DOCTEUR ?

L'accroissement en taille de MINIX est dû exclusivement à la présence d'un grand nombre de nouvelles commandes et de leurs fichiers source. Jugez plutôt : 4 Mo de fichiers binaires ! La grande majorité de ces commandes ont été écrites par des utilisateurs de MINIX. Il y a de tout : des petits jeux (Tic-Tac-Toe) comme des commandes faisant cruellement défaut dans la version 1.1 : ps, ou process-status, entre autres (enfin !). Citons encore un débogueur symbolique, un programme de backup, une implémentation du langage Awk, un éditeur de disques, les programmes de communication Kermit et Zmodem et pas moins de trois nouveaux éditeurs de texte ! Il y a Sed, l'éditeur batch standard d'UNIX, Elle, un compatible emacs, et Elvis, un compatible vi. Avec ça, on ne peut plus se plaindre de ne pas trouver éditeur à sa convenance. Le compilateur C est toujours aussi peu brillant par sa rapidité et la qualité du code produit. Un regret en passant : toujours pas de librairie mathématique livrée en standard, malgré la présence d'un encourageant float.h. Il semblerait néanmoins que la version 2.0 du compilateur (norme ANSI) soit en cours de finalisation.

Ces détails mis à part, le langage C est bien implémenté, et la librairie C-UNIX standard est complète. Quelques "plus" par rapport à la version précédente : une librairie graphique BSD curses, et l'adaptation de certains fichiers d'inclusion à la norme POSIX.

Le système d'exploitation lui-même a été grandement retravaillé. Les signaux et les appels système ont été revus et complétés, et les entrées/sorties optimisées. MINIX a gagné en rapidité et en robustesse, et s'est nettement rapproché d'UNIX pour tout ce qui concerne le fonctionnement et la richesse des commandes disponibles (plus de 175 !).

La documentation est bien écrite et donne un descriptif de toutes les commandes, avec des détails supplémentaires sur les plus complexes, comme le débogueur et les éditeurs de texte. Un chapitre est consacré à la présentation de MINIX, avec un premier aperçu du système de gestion de fichiers, de la notion de processus et une visite guidée dans les répertoires, une fois l'installation complétée. Le tout occupe environ 300 pages d'un solide classeur au format US, les 400 pages restantes étant consacrées aux principaux fichiers source du système. Les fichiers présents dans cette partie du manuel forment la partie commune des versions de MINIX tournant sur les 4 machines citées plus haut : un bel exemple d'universalité, car l'on constate qu'il s'agit de l'intégralité du système de gestion de fichiers et du gestionnaire de mémoire ! Une bonne partie du source du noyau est également reproduite, dans sa version PC.

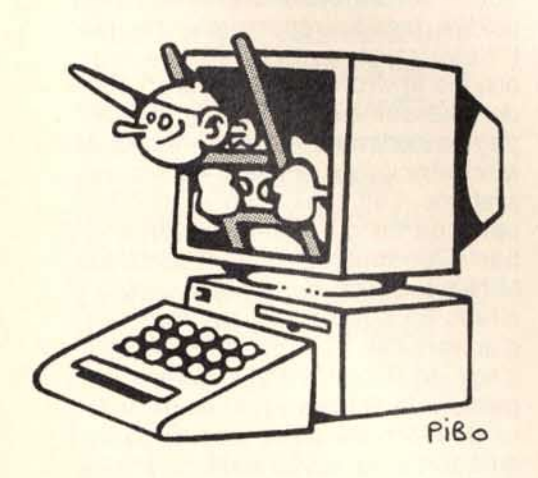


CONCLUSION

Il est difficile de faire le tour des fonctionnalités de MINIX en une seule fois. Une utilisation prolongée permettra de mieux appréhender ses possibilités et de faire apparaître ses éventuelles faiblesses. Globalement, le bilan est positif : MINIX a beaucoup mûri depuis sa première version sur ST, et il semble qu'il n'ait pas fini d'évoluer. Actuellement, plusieurs milliers de personnes participent au phénomène MINIX dans le monde entier et l'impulsion donnée semble être suffisante pour garantir à ce système un bel avenir.

Malgré la relative modestie d'une configuration ST, UNIX sur ST est devenu une réalité. En attendant la disponibilité (lointaine...) de la version Atari d'UNIX system V version 4 sur TT, vous pourrez vous initier au multitâche à peu de frais : MINIX-ST 1.5 coûte actuellement moins de 1400 francs à la librairie "Le Monde en Tique". Nous ne reviendrons pas sur les raisons qui pourraient vous pousser à étudier MINIX de plus près (voir ST Mag numéro 47). Disons seulement que MINIX n'est pas un jouet, et que son utilisation nécessite un certain apprentissage, et un peu de travail. Pour ceux (et celles) qui s'investiront dans cette exploration, le jeu en vaut largement la chandelle et nous serons là pour les guider, au fur et à mesure de nos prochains travaux pratiques.

Nicolas Spengos



MINIX : SUBTILITÉS D'INSTALLATION

D'un point de vue "puriste", lorsque l'on utilise un certain système d'exploitation sur une machine, il est préférable d'avoir booté sur ce système. Malheureusement, dans ce cas, les utilisateurs de disques durs configurés comme systèmes de démarrage sont obligés de rivaliser de vitesse avec la procédure de boot en appuyant frénétiquement sur la touche ALT. Que dire alors de ceux qui, comme l'auteur de ces lignes, sont allergiques à la prolifération de disquettes de boot correspondant à des environnements différents. Parmi la foultitude de programmes qui accompagnent MINIX, un seul répond au nom de MINIX.PRG : il sert tout bonnement à lancer la version exécutable du système (fichier image) à partir du bureau GEM. Cela marche bien, à un détail près : MINIX.PRG ne reconnaît pas les fichiers images de MINIX qui ont été modifiés pour prendre en compte le clavier français. C'est bien un comble, car cette modification (effectuée avec FIXKEYS.PRG) est décrite dans le manuel ! MINIX.PRG n'accepte que les versions du fichier image sorties tout droit du compilateur, sans être passées par une quelconque bidouille. Il faut donc aller modifier les codes du clavier directement dans le source de MINIX. Ces codes sont définis dans le fichier /usr/src/kernel/keymap.h ; pour obtenir un nouveau fichier keymap.h, il suffit d'utiliser le programme KEYTBL.TTP qui se trouve sur la disquette d'utilitaires TOS de MINIX (ex. keytbl key.h). Il ne reste plus ensuite qu'à transférer ce fichier vers MINIX avec la commande "tos -ra /dev/dd0 key.h > keymap.fra.h", de modifier le fichier makefile pour qu'il prenne en compte keymap.fra.h, et de lancer la commande make. Etant donné qu'il faut tout recompiler, cela risque d'être assez long, aussi est-il conseillé de suivre scrupuleusement les indications du manuel concernant la construction du système. Si vous hésitez à vous lancer sans plus d'explications, vous trouverez en téléchargement, sur le serveur 3615 STMAG, ainsi que sur la disquette du magazine, les matériaux nécessaires à cette opération, avec en prime de quoi changer la fonte écran de MINIX (fonte ST standard...). Vous pouvez aussi m'écrire en Bal MINIX, pour toutes vos questions.

Pour les utilisateurs de disques durs, il peut également être intéressant de "démonter" les partitions MINIX, afin d'empêcher les programmes TOS d'y accéder. En effet, ne pas installer ces partitions sur le bureau GEM ne suffit pas : elles seront toujours accessibles aux applications. Le petit programme C suivant, à placer dans le dossier AUTO, permet de remédier à ce problème :

```
#include <stdio.h>
#include <osbind.h> /* pour Lattice C5, ou tos.h pour Turbo C2*/

#define DRVMAP     xxL/* entier long codant les disques logiques que  */
                      /* vous voulez conserver ; en binaire, le bit 0 */
                      /* correspond au lecteur A, le bit 1 à B, ...   */
                      /* mettez à 0 les bits correspondants aux       */
                      /* partitions MINIX                             */
int main(void)
{   void *oldssp, *stack;
    unsigned long *_drvbits;

    stack = NULL;
    oldssp = Super(stack);          /* passer en mode superviseur 68000 */
    _drvbits = (unsigned long *) 0x4C2;
    *_drvbits = DRVMAP;
    puts("Partitions Minix démontées\n");
    Super(oldssp);
}
```



PROGRAMMATION DU SON SUR STE

Les ordinateurs de la famille STE peuvent reproduire des sons digitalisés sans passer par le 68000, grâce à un DMA sonore dont nous allons étudier la programmation. Cet aperçu de la gestion des échantillons sur STE devrait faciliter un habillage sonore luxueux (donc indispensable) de vos applications et démos préférées.

UNE PETITE MINUTE...

Avant d'entrer dans le vif du sujet, il est nécessaire d'expliquer quelques notions qui nous seront utiles par la suite. Dans le "Sampling" (échantillonnage), contrairement à l'analogique qui manipule des grandeurs continues (tensions électriques) et use de mécaniques aux résultats fluctuants (frottement du saphir sur le vinyle ou défilement de la bande magnétique sur les têtes du magnétophone), on délaisse l'imprécision par l'entremise du numérique, pour coder et transmettre les informations en leur donnant des valeurs sans équivoque : soit 0 soit 1. Mais qu'échantillonne-t-on au juste ?

Tout phénomène sonore génère des variations de pression de l'air, se propageant comme des ondes sur la surface de l'eau. Un son dégage donc à un instant donné une force précise, une intensité mesurable. La technique d'échantillonnage consiste, par l'entremise d'un convertisseur analogique-digital (ADC), à prélever et à numériser à intervalles très courts et réguliers des échantillons de ces intensités.

Le STE n'est pas pourvu d'un ADC; mais on trouve aisément des échantillons tout faits, et nombre d'éditeurs de logiciels proposent des échantillonneurs pour ST et STE (cartouche ADC plus logiciel d'édition) à des prix raisonnables. Le STE est en revanche équipé d'un convertisseur digital-analogique 8 bits, propre à retraduire de l'information numérique (des échantillons) en signal audio écoutable. Les 2 sorties RCA mono du STE tiennent ce signal à disposition pour écoute sur un périphérique de votre choix (chaîne Hi-Fi). Dernier avantage de l'échantillon numérique : l'information, facilement stockable sur disque ou disquette, est manipulable à volonté en mémoire RAM.

La qualité sonore du son échantillonné variera, dans certaines limites (performances des composants électroniques utilisés, capacités d'écoute de l'auditeur, etc.) selon deux facteurs : le domaine de fréquences dans lequel l'échantillon est enregistré et la résolution de l'échantillon, déterminée par la finesse de la grille de valeurs disponible lors du codage : le nombre de bits sur lesquels l'information est codée (échantillonnage sur 8 bits : 256 nuances possibles).

Les données digitales sont stockables en mémoire, mais à quel prix ! Le temps que durera l'échantillonnage se payera cher en mémoire : gare au choix des fréquences (de 20Hz à 48.000Hz), ainsi qu'à la finesse de résolution.
Pas de panique ! 512 ko de Ram suffiront quand même à satisfaire les convertisseurs 8 bits du STE pour des applications courantes (démos) Mais comme les STE sont pourvus de convertisseurs 8 bits, le STE travaille avec une table de sonorités codée sur 8 bits (2^8=256), et nous resterons loin de la consommation des lecteurs lasers et de certains instruments de musique digitaux (16 ou 24 bits à 48 kHz). Nos lecteurs pourront encore se reporter à notre numéro spécial musique de septembre 1990, pour plus de détails sur l'échantillonnage et le Midi ; mais vous en savez assez à présent pour aborder un travail plus concret.

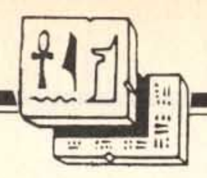
LE SYSTEME PCM (PULSE CODE MODULATION)

Vous avez dû déjà en entendre parler, ne serait-ce qu'en musique avec les synthétiseurs, boîtes à rythmes et autres machines à sons. Notons ici, sur STE, une particularité : les données sont codées sur 8 bits, certes, mais leurs 256 valeurs sont ordonnées de -128 à 127, alors que celles produites par ST REPLAY, par exemple, vont de 0 à 255. Le format PCM est en fait caractérisé par une position de repos centrale (0).

Quelques conseils pratiques : quand vous voudrez jouer un échantillon provenant du logiciel ST Replay, il faudra en conséquence soustraire la valeur 128 à tous les octets du fichier. Quelques curieux tenteront cependant de jouer un fichier SPL par le DMA sonore sans avoir converti les données (baissez le son !) ; vous reconnaîtrez sans doute votre échantillonnage, quoique frappé de nombreuses distorsions.

PRINCIPE DU SON PCM DE LA FAMILLE DES STE

Voyons maintenant le parcours de notre "sample". Les échantillons du STE sont donc chargés en mémoire ; le DMA ira chercher, à un intervalle de



temps régulier imposé, un octet du "sample" pour le convertir en un événement sonore grâce au convertisseur DAC (Digital to Analog Converter), sans utiliser le 68000 : pas de perte de temps de calcul 68000 pour vos démos ou applications. Le DAC travaille à des fréquences prédéfinies, qui sont approximativement 50, 25, 12,5 et 6,25 kHz. Une fois converti, le son est filtré à une fréquence égale à 40 % de celle de l'échantillonnage par un filtre passe-bas ; une opération qui permet d'obtenir un son de meilleure qualité, avec une bonne réduction du souffle. Le signal passe ensuite par un autre filtre passe-bas (le National LMC1992 Volume/Tone Controller), travaillant à une fréquence fixe de 16 kHz.

Le signal est, après cela, prêt à être envoyé aux fiches RCA (connectez votre STE sur votre amplificateur en entrée "aux") ou moniteur (en mono) par la prise Péritel. Il faut noter que le DMA sonore peut aller prendre soit deux octets (sample en stéréo) soit un octet (mono). Il est aussi pos-

DESCRIPTIONS DES REGISTRES

```
FF8900 — — — — — -  CC RW  REGISTRE DE CONTROLE DU DMA SON
                    CC:
                          00= DMA SONORE DÉSACTIVÉ (ETAT INITIAL)
                          01= JOUE LE SAMPLING SONORE UNE FOIS
                          11= REPETE LE SAMPLING A L'INFINI

FF8902 — — — 00XX XXXX RW OCTET FORT DE L'AD. DE BASE
FF8904 — — — XXXX XXXX RW OCTET MOYEN DE L'AD. DE BASE
FF8906 — — — XXXX XXXX RW OCTET FAIBLE DE L'AD. DE BASE

FF8908 — — — 00XX XXXX RO OCTET FORT DE L'AD. DU COMPTEUR
FF890A — — — XXXX XXXX RO OCTET MOYEN DE L'AD. DE COMPTEUR
FF890C — — — XXXX XXXX RO OCTET FAIBLE DE L'AD. DE COMPTEUR

FF890E — — — 00XX XXXX RW OCTET FORT DE L'AD. DE FIN
FF8910 — — — XXXX XXXX RW OCTET MOYEN DE L'AD. DE FIN
FF8912 — — — XXXX XXXX RW OCTET FAIBLE DE L'AD. DE FIN

FF8920 — — — M000 00FF RW CONTROLE DU MODE SONORE
                   M:    0     MODE STEREO (ETAT INITIAL)
                         1     MODE MONOPHONIQUE
                   FF:   00    FREQUENCE= 6258 Hz (ETAT INITIAL)
                         01    FREQUENCE=12517 Hz
                         10    FREQUENCE=25033 Hz
                         11    FREQUENCE=50066 Hz
```

CHAQUE MOIS

Recevez chez vous ST MAGAZINE

Le magazine de vos programmations, de vos jeux, et de vos applications PAO et graphiques.







Abonnez-vous à ST MAGAZINE

et profitez de conditions exceptionnelles d'abonnement !

11 NUMÉROS POUR 225 FRANCS!

22 NUMÉROS POUR 400 FRANCS!

OUI, je souhaite m'abonner à ST MAGAZINE pour:

❑ 11 numéros au tarif de 225 Francs, soit 2 numéros gratuits.

❑ 11 numéros au tarif de 625 Francs, soit 2 numéros gratuits et je reçois la disquette ST MAGAZINE, qui contient tous les listings publiés.

❑ 22 numéros 400 Fancs, soit 6 numéros gratuits.

❑ 22 numéros au tarif de 900 Francs, soit 6 numéros gratuits et je reçois la disquette ST MAGAZINE, qui contient tous les listings publiés.

Date : Signature : (des parents pour les mineurs)

Je paye ci-joint par :

❑ Chèque bancaire ou postal à l'ordre de **PRESSIMAGE**

❑ Mandat postal pour l'étranger.

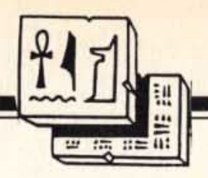
Mme ❑ Mlle ❑ M^r ❑

NOM : ____________ Prénom : ____________

Adresse : ____________

Code postal : ____________ Ville : ____________ tél : ____________

Bulletin à retourner sous enveloppe affranchie à ST MAGAZINE ABONNEMENTS 19 rue Hégésippe Moreau 75018 PARIS



sible de mixer le son provenant du processeur sonore (Yamaha 2149) à l'échantillon (sur les deux canaux sonores) grâce au National LMC1992.

Reportons-nous à présent à l'encadré décrivant les registres du DMA son : en commentaires, vous trouverez l'adresse de début (ad.de base) et l'adresse de fin (ad. de fin) du sample. Les registres du compteur indiquent à chaque instant l'endroit où se trouve le DMA dans le sampling.

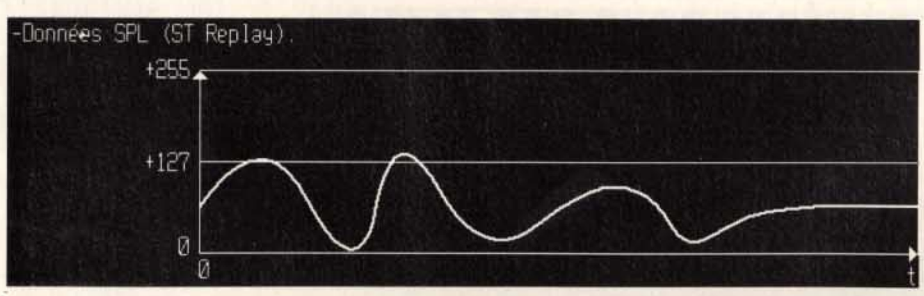
COMMENT JOUER UN ÉCHANTILLON ?

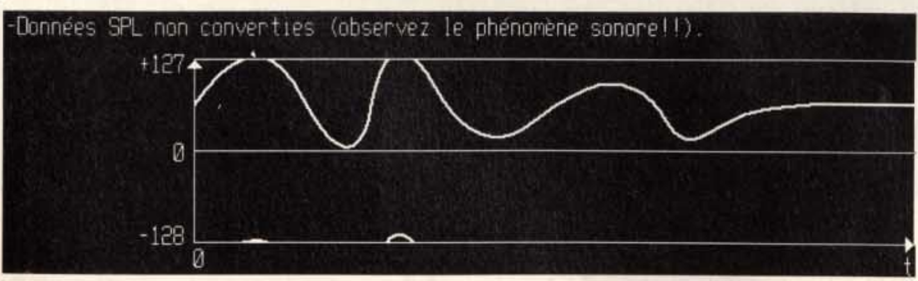
Le moyen le plus simple de jouer un échantillon sonore par le DMA est de charger le sample en mémoire centrale, et de le convertir, si c'est un fichier du type SPL (ST replay), en soustrayant à chaque octet 128, puis d'écrire l'adresse de départ dans les registres appropriés : $FFFF8903.b pour l'octet de poids fort, $FFFF8905.b pour l'octet de poids moyen, $FFFF8907.b pour l'octet de poids faible ; l'extension .b n'est pas une syntaxe exacte, nous l'utilisons seulement pour vous rappeler qu'un octet doit être écrit à un adresse impaire. Il faut ensuite écrire l'adresse de fin de l'échantillonnage dans les registres DMA correspondant ($FFFF890C.b pour l'octet fort...).

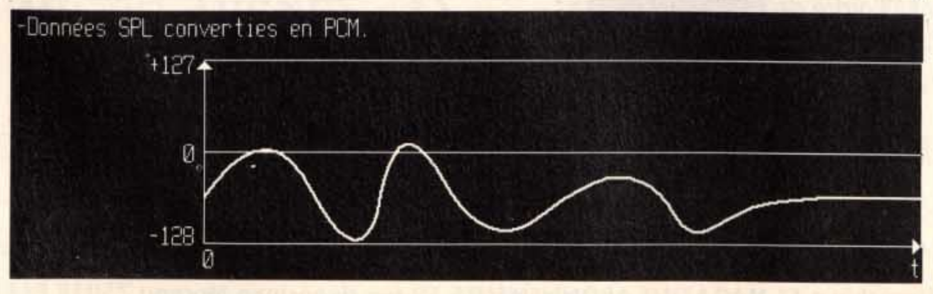
En fait, l'ordre dans lequel on écrit les adresses de départ et de fin importe peu, mais n'allez pas écrire l'adresse de départ dans les registres DMA d'adresse de fin de sampling ou vice versa, vous joueriez toute la mémoire de votre machine sauf votre sampling! Une fois que vous avez indiqué au DMA l'endroit où se trouve votre échantillonnage, il faut lui préciser à quelle fréquence et en quel mode (mono ou stéréo) il devra jouer grâce à l'adresse $FFFF8920.
Pour lancer le processus de restitution sonore, il faut indiquer au DMA s'il doit jouer l'échantillonnage une fois ou bien à l'infini. Nous vous laissons consulter le tableau de description des registres du DMA sonore, pour connaître les diverses possibilités de programmation de ce dernier.

DES MILLE ET UNE FAÇONS DE JOUER LES ÉCHANTILLONS

La méthode donnée plus haut est la plus simple et la plus rapide à programmer. On peut aussi jouer des assemblages d'échantillons, c'est ce nous allons voir à présent.
Le DMA sonore produit un signal (le "DMA sound active"), qui est à 1 quand le chip joue et à 0 quand il ne joue pas (ou plus !). Quand un échantillonnage se termine en mode de répétition (3 dans $FFFF8901.b), il y a un passage pour le "DMA sound active" de 1 à 0 puis un retour à 1 lorsque le sampling reprend. Ce signal est une entrée externe du Timer A du MFP 68901.







Quand le DMA sonore est lancé en mode répétition, les adresses de départ et de fin sont rangées dans une zone inchangeable par programmation, ce qui permet d'écrire l'adresse du prochain échantillon pendant que l'actuel est joué. Ce système nous ouvre de grandes possibilités pour jouer des assemblages d'échantillons sonores. En fait, la zone mémoire ("holding area") qui garde les adresses de départ et de fin est rechargée lorsque le sampling commence ou reprend ; cela permet d'enchaîner des échantillons différents. Cette méthode est assez simple à programmer dans n'importe langage.

Voici, à présent, un exemple de programmation du DMA sonore avec TIMER A pour jouer une succession d'échantillons. Prenons par exemple trois échantillons A, B, et C. Pour jouer A deux fois, B quatre fois, et pour finir C une fois, il faut procéder dans l'ordre comme suit :
- Positionner le TIMER A en mode Event Count mode avec son compteur à 1 ;
- Charger les adresses de début et de fin de l'échantillon A dans les registres du DMA sonore ;
- Positionner le DMA sonore en mode Répétition en écrivant un 3 dans le registre de contrôle ($ffff8900) ; la routine d'interruption du timer A est lancée au moment où l'échantillon commence sa troisième répétition ;
- Charger les adresses de début et de fin de l'échantillonnage B dans les registres du DMA sonore ; à la fin de la troisième répétition, elles passeront dans les véritables registres et le DMA jouera le sampling B ;
- Positionner le compteur du TIMER A à 4 puis quitter l'interruption jusqu'à la prochaine interruption. A chaque répétition, le compteur du TIMER A est décrémenté jusqu'à ce qu'il atteigne –1, et l'interruption se déclenche ;
- Écrire les adresses de début et de

fin de l'échantillon C dans les registres DMA sonore ;

- Positionner le compteur du timer A à 0. Quand il est -1, on peut si l'on désire recommencer la série ou alors stopper la restitution sonore en désactivant le timer en question ;

Comme vous avez pu le voir, il faut positionner le compteur d'événements du timer A à un nombre N-1 pour obtenir N restitutions ; cela est dû au fait que le timer ne se déclenche qu'au moment où son compteur vaut -1 (255 sur un octet). Vous avez pu remarquer la souplesse qu'offre le DMA sonore : le "Holding Area" (tampon) s'avère très pratique. Grâce à ce dernier vous pourrez jouer des samplings en donnant l'impression de non-rupture sans problème ou jouer vos Soundtracks comme sur AMIGA !

Le listing ci-joint a été développé sous Devpac, mais il est volontairement simplifié pour ne pas risquer de rebuter les jeunes programmeurs qui nous rejoignent. Nous reprendrons d'ailleurs bientôt le chapitre des réjouissances sonores sur STE, lors des débuts d'une série d'articles leur étant tout spécialement consacrés.

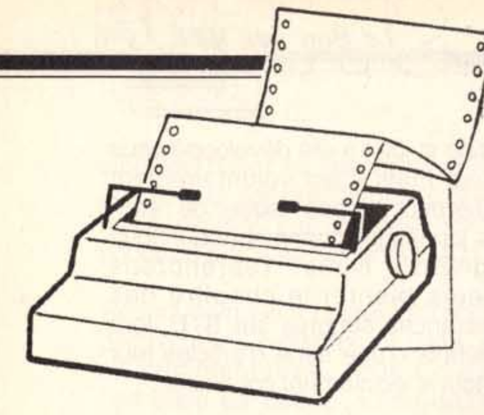
The Constellations (CST)
Elysee Phoenix ADE

```
        *** Son PCM #1 pour Ste,TT par CST ***
Startsound :
    pea         0               * Passage en mode Superviseur
    move.w      #$20,-(sp)      * car les registres du DMA
    trap        #1              * sonore ne sont accessibles
    addq.l      #6,sp           * qu'à cette condition.
    move.b      #0,$ffff8901.w          *On stoppe le DMA
    move.w      #0,-(sp)                *On ouvre le fichier
    move.l      #name,-(sp)
    move.w      #$3d,-(sp)
    trap        #1
    add.l       #8,sp
    tst.l       d0              *Si d0<0,
    bmi         out1            *on s'en va, puis on stocke
    move.w      d0,nlogl        *le num. logique du fichier;
    move.w      #2,-(sp)                *On recherche
    move.w      nlogl,-(sp)             *la taille du
    move.l      #0,-(sp)                *fichier
    move.w      #$42,-(sp)
    trap        #1
    add.l       #10,sp
    move.l      d0,length               *On le stocke
    move.w      nlogl,-(sp)
    move.w      #$3e,-(sp)
    trap        #1
    add.l       #4,sp
    move.w      #0,-(sp)
    move.l      #name,-(sp)
    move.w      #$3d,-(sp)
    trap        #1
    add.l       #8,sp
    tst.l       d0
    bmi         out1
    move.l      #music,-(sp)            *On charge le
    move.l      length,-(sp)            *fichier à
    move.w      nlogl,-(sp)             *partir de
    move.w      #$3f,-(sp)              *la zone
    trap        #1                      *music
    add.l       #12,sp
    move.w      nlogl,-(sp)             *On referme
    move.w      #$3e,-(sp)              *le fichier
    trap        #1
    add.l       #4,sp
set_pcm
    lea         music,a1                *Conversion SPL
    lea         (a1),a2                 *à PCM
    add.l       length,a2
pcml
    move.b      (a1),d0                 *On soustrait
    sub.b       #128,d0                 *la valeur 128
    move.b      d0,(a1)+                *à chaque octet
    cmp.l       a2,a1                   *jusqu'à la fin
    blt         pcml                    *du sample
    move.l      #music,snda             *On va écrire
    lea         snda,a0                 *l'adresse
    jsr         setsnd2                 *de départ
    lea         music,a1                *puis
    add.l       length,a1               *l'adresse
    move.l      a1,snda                 *de fin
    move.l      #snda,a0                *du sample
    jsr         setsnd3
    move.b      #128+3,$ffff8921.w  *On règle la fréquence
                                    *de restitution
    move.b      #3,$ffff8901.w      *C'est parti
                                    *en mode répétition
    move.w      #7,-(sp)
    trap        #1

    addq.l      #2,sp
out1
    move.w      #0,$ffff8900.w
    clr.l       -(sp)
    trap        #1

setsnd2
*Sous-prg de chargement de l'ad. de départ contenue ds (a0)
    move.b      3(a0),$ffff8907.w   *Octet faible
    move.b      2(a0),$ffff8905.w   *Octet moyen
    move.b      1(a0),$ffff8903.w   *Octet fort
    rts
setsnd3
*Sous-prg de chargement de l'ad. de fin contenue dans (a0)
    move.b      3(a0),$ffff8913.w   *Octet faible
    move.b      2(a0),$ffff8911.w   *Octet moyen
    move.b      1(a0),$ffff890f.w   *Octet fort
    rts
name
    dc.b        'd:vega.spl'  *Ici préciser le lecteur et le
                              * nom du fichier à jouer
            section   bss
snda    ds.l        1
nlogl   ds.w        1
length  ds.l        1         *Pour un sampling + grand que
music   ds.b        1024*256  *256 Kb,changer taille buffer

        end
```



MOYENS INFORMATIQUES POUR AVEUGLES : OU EN EST-ON ?

L'évolution des matériels et logiciels vers des interfaces graphiques de plus en plus sophistiquées, sollicitant constamment l'usage de la vue, permet à un public de plus en plus nombreux d'accéder à des outils dont l'apprentissage est facilité. Mais qu'en est-il de l'utilisation de l'outil informatique, pour aider à l'insertion des personnes qui souffrent d'un handicap majeur face à un écran : les aveugles et malvoyants ?

Les aveugles souffrent, comme les autres handicapés, de problèmes d'insertion, en particulier dans le monde du travail. Le sens du toucher a permis, grâce aux travaux de Louis Braille, de permettre aux aveugles une communication par l'écrit, par l'invention d'un alphabet qui porte son nom, formé de points en reliefs sur un papier épais. Pourtant ce handicap face au travail se pose de plus en plus, au fur et à mesure que l'informatique prend une part grandissante dans l'activité humaine. Et là, le cas des aveugles est bien évidemment le plus crucial. Les outils principaux d'interaction entre l'homme et la machine restent, pour le moment, le clavier, qui permet de transmettre une information à l'unité de traitement, et l'écran, qui permet d'informer l'utilisateur de la réaction du système à cette information. A ces outils s'ajoutent bien évidemment l'unité de stockage de l'information, qui n'interagit pas directement avec l'utilisateur, et l'unité d'impression, qui permet de garder une trace écrite de l'information traitée. De nouveaux moyens de communication de l'homme vers la machine sont peu à peu apparus, comme les tablettes graphiques, et surtout les systèmes graphiques à pointeurs (souris, track-ball...) qui tiennent une place de plus en plus importante dans les systèmes interactifs, faisant un appel constant au sens certainement le plus sollicité par un ordinateur : la vue. Paradoxalement, l'ordinateur est en train de devenir un formidable outil d'insertion des non-voyants dans le monde du travail.

Le PC, principal système accessible aux non-voyants

Le principal handicap des micro-ordinateurs de type PC vis-à-vis du public, face à l'apparition des interfaces graphiques popularisées par les Macintosh, Atari ST et Amiga, était le côté rébarbatif d'un système MS-DOS dont le seul outil interactif était le clavier; plus tard, seulement apparurent les surcouches GEM (Graphics Environment Manager) et Windows. Mais c'est justement cette caractéristique de MS-DOS qui amena le développement sur PC de systèmes adaptés aux non-voyants, leur sens déficient s'accomodant volontiers du manque de graphismes et d'icones.

De plus, ces systèmes sont certainement les plus faciles à adapter aux non-voyants, en raison de la simplicité d'ajouts de cartes spécifiques et de leur coût très bas : il est maintenant courant de trouver des PC-AT à moins de 10.000 francs, avec une unité de stockage de masse de plusieurs dizaines de millions de caractères.

Les imprimantes, périphériques privilégiés

Pour ces périphériques, l'adaptation aux non-voyants a été on ne peut plus plus rapide, aussi les imprimantes Braille, bien que très chères, sont assez répandues, sous deux formes. La première, de loin la plus répandue est l'embausseuse : c'est l'équivalent d'une imprimante matricielle, où la tête d'impression est formée d'une matrice de 8 points (2 sur 4), qui va déformer le papier, créant le caractère Braille en relief. Précisons que le caractère Braille littéraire est formé d'une matrice de 6 points (2 sur 3), créant un jeu de 64 caractères (6 bits

a b c d e f

g h i j k l

m n o p q r

s t u v w x

y z

les points fins ne sont là que pour rendre la lecture plus facile par les voyants

- ou unités élémentaires - d'information), et qu'un jeu Braille informatique (2 sur 4 points) a été imaginé, largement inspiré du précédent, puisque la représentation des 64 caractères de base a été conservée, à laquelle on a ajouté des caractères utilisant les deux points supplémentaires, de manière à représenter les 256 caractères ASCII (American Standard Codes for Information Interchange, norme de codage de caractères adoptée par tous les fabricants de matériels informatiques), condition nécessaire pour permettre aux aveugles d'accéder à la programmation.

Le deuxième type d'imprimante est apparu très récemment, grâce à la technologie d'impression par jet de cire (encre projetée sur le papier), qui permet à une imprimante destinée aux voyants d'imprimer des caractères Braille sous forme de points noirs en relief. A notre connaissance, le premier modèle a été fabriqué par la firme Dataproduct, qui contient, en standard, un jeu de caractères Braille parmi les jeux proposés. Le premier avantage de cette technique est de permettre à un utilisateur d'utiliser la même imprimante pour produire des documents en noir et en Braille. Un autre avantage est de permettre l'utilisation de papiers traditionnels par opposition aux embausseuses qui nécessitent d'employer un papier épais (au moins 170 grammes par mètre carré) pour produire un point de bonne qualité et résistant à l'usure. Par contre, un inconvénient vient contrebalancer ce qui précède : c'est le coût de la cire employée par cette imprimante.
Conséquence du principe de fonctionnement de ces imprimantes, elles peuvent être gérées par tout logiciel permettant une impression en mode non graphique comme la plupart des traitements de textes sur PC et ST. Il suffit simplement de régler les paramètres de la page par rapport aux caractéristiques d'une page Braille. Quelques contraintes de saisie demeurent cependant : elles concernent les majuscules et les nombres.

Remplacer la vue : synthèse vocale et écran Braille

La communication avec le système informatique se faisant essentiellement par l'intermédiaire du clavier et de l'écran, il apparaît difficile aux non-voyants d'y accéder. Heureusement, il existe certains dispositifs permettant d'améliorer le dialogue entre le non-voyant et le système informatique. Le premier dispositif est un clavier modifié, dont les touches sont moulées de manière à former en relief le caractère Braille correspondant, permettant ainsi à faible coût l'entrée de caractères.
Reste alors le problème du retour de l'information vers l'utilisateur, à des fins de vérification. Ce retour peut être effectué à l'aide d'un dispositif basé sur la synthèse vocale. Chaque caractère entré par l'utilisateur, et chaque mot affiché par l'ordinateur, en mode texte, grâce à un dispositif qui prononce alors chaque caractère ou chaque phonème : la carte vocale détourne en quelque sorte l'affichage du système. Ces dispositifs coûtent plusieurs milliers de francs, et restent peu accessibles pour le moment. Il en existe maintenant des versions adaptées au Minitel, rendant ainsi possible l'usage de ce merveilleux outil. Le dernier dispositif est quant à lui nettement plus onéreux : il s'agit d'un écran Braille dit éphémère qui permet d'afficher de une à plusieurs

LE GRAND RETOUR DE L'AVENTURIER FOU

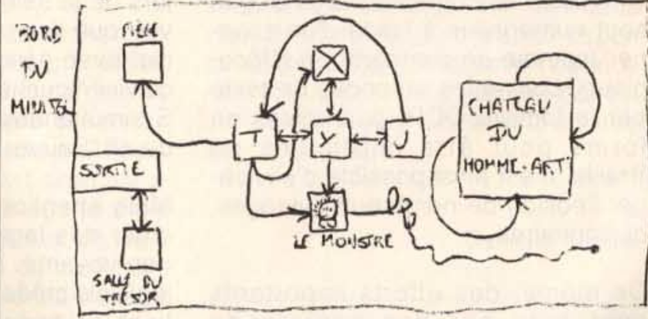
Ce mois-ci, la solution du jeu "Space Quest 12: the Arnak of the Sans Tury". Je vous rappelle que vous pouvez m'envoyer vos propres soluces à:

L'AVENFOU
8 allée Voufer-vouar
75942 PARIS CEDEX 23

Vous pouvez aussi me contacter sur le 3615 ABCDEFGHIJKLMNOPQRSTUVWXYZ.
Yeah, man, un petit truc de l'Avenfou pour commencer: wouah, djeune kide, si tu es sur Paris et que tu voudrais appeler en Province, l'astuce consiste à presser rapidement sur 1 puis 6! Si tu n'entends pas de DJUUUUUU, le truc n'a pas marché; recommence!

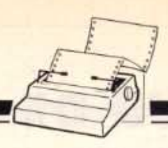
Go North, Take Sword, Go North, Go South, Drop Sword, Go South, Go North, Take Sword, Go West, Give Shruberry To The Knights Who Say Niii, Throw Sword In The Teeth of The Jealous Princess and Kiss the Monster. Open Minitel, Type 3,4,6,6,0,3,5,7, Say "Hello" to Oxygene in RP. Type 7,2,0,0,9,2,3,9, Say "Hi Cucumber!" (4 channels) in Lyon. Type 4,8,6,5,9,2,7,3, Say "Hello!" to Coca in Paris. Type 3,9,6,2,6,7,5,9 and Say "Hi!", to Kevius from Top Secret, in Yvelines. Type 3,4,1,3,5,3,1,5 and Say "KooKoo" to TelDev in Paris. Type 2,0,9,3,3,7,0,5 and Say "Pritou" to Logarythm in Province.

Voici un plan envoyé par un Aventurier-lecteur, Jérôme Hanichelle qui déménage souvent, dommage, je ne vais pas pouvoir l'abonner (gagnez un abonnement en envoyant une soluce à l'adresse donnée ci-dessus! L'Avenfou est cool!), c'est le plan pour accéder à la touche Y sans perdre de points d'énergie. Merci, Jérôme!



Une dernière astuce: pour augmenter ton score sur le 3615, appuie sur la touche Sommaire avant de taper STMAG, tu verras ton score s'accroître de minute en minute. A un million, tu as une vie supplémentaire!

On se retrouve le mois prochain, hein, comme d'habitude?



lignes sur jusqu'à 80 caractères. Un clavier une ligne de 80 caractères coûte en effet 80.000 francs ! Le principe en est simple : l'écran, qui est par exemple placé sous le clavier est formé de rangées de trous placés selon les normes d'espacement Braille. Lorsqu'un caractère est affiché à l'écran, des picots plastiques sortent des cavités, générant en relief le caractère Braille correspondant. Le coût d'une station formée d'un PC, d'une carte de synthèse vocale, d'un clavier et d'un écran Braille dépasse la centaine de milliers de francs... Elle permet néanmoins à tout non-voyant d'accéder à des emplois de saisie, programmation...

La traduction du Braille vers le noir et sa réciproque

Le problème de la traduction du noir (impression traditionnelle) vers le Braille se pose. Il existe des systèmes de saisie de texte pour non-voyants, sous la forme de machines à écrire interfacées avec des imprimantes matricielles traditionnelles, qui sont nettement moins onéreux que les systèmes informatiques. Un non-voyant peut alors taper une lettre, et l'imprimer en noir, pour pouvoir l'adresser à un voyant. Mais en ce qui concerne les ouvrages, journaux, il est impératif de pouvoir employer des moyens automatiques de traduction de textes imprimés en noir en Braille. C'est aujourd'hui possible en raison des progrès de la reconnaissance optique de caractères (en anglais OCR pour Optical Character Recognition). Les pages sont numérisées à l'aide d'un scanner (comme en comporte un télécopieur), converties en codes de texte par le logiciel OCR, puis mises en forme pour être imprimées en Braille. Il est ainsi possible d'envisager l'édition de nombreux ouvrages, dictionnaires...

De même, des efforts importants sont faits par des centres de recherche en France ou à l'étranger, pour permettre la traduction automatique en noir de documents Braille déjà existants. A partir de moyens optiques (scanner, caméra vidéo), les images sont numérisées, filtrées, puis analysées, et converties en codes de texte au standard ASCII. Les documents écrits en Braille avec les moyens du bord (c'est-à-dire le plus souvent à l'aide d'un poinçon) peuvent donc être publiés à destination des voyants.

Les interfaces graphiques

Nous avons abordé plus haut les problèmes générés par le développement d'interfaces graphiques de plus en plus sophistiquées. Celles-ci sont inaccessibles aux non-voyants puisqu'elles nécessitent une utilisation intensive de pointeurs graphiques qui se déplacent, suivant les gestes de la main, et que l'on doit constamment suivre des yeux.

Les logiciels sont dans un mode où ils attendent la venue d'un événement (déplacement du pointeur graphique, "clic" sur un bouton du pointeur, ou frappe d'une touche du clavier). Le premier moyen que l'on peut envisager pour rendre possible l'utilisation de tels logiciels par des non-voyants serait que les développeurs (ou programmeurs) fassent un effort pour adapter leurs produits. Il est par exemple envisageable de doubler toutes les options des menus déroulants par ce que l'on appelle généralement des raccourcis clavier (combinaison de touches du clavier produisant le même effet que la sélection d'un objet dans un menu déroulant), et ensuite de relier le module du programme qui réagit à l'événement à une carte de synthèse vocale, de manière à prononcer l'option choisie. Par exemple, lors de la sélection de l'option "Sauver" que l'on retrouve dans tout logiciel avec généralement le raccourci clavier "touche CONTROL et touche S simultanées", l'ordinateur prononcerait "Sauver un fichier".

Mais attention ! Il ne s'agit pas de créer des logiciels spécifiques pour non-voyants, il s'agit d'adapter des logiciels créés de manière à être utilisables par une personne qu'elle soit voyante ou non : pour pouvoir s'insérer dans le marché du travail, un non-voyant doit pouvoir utiliser un produit standard, afin de pouvoir travailler en osmose avec ses collègues, utilisateurs des mêmes logiciels. C'est une condition impérative à la réussite de l'insertion au travail !

Bien sûr, certaines opérations ne peuvent être résumées par un raccourci-clavier, en particulier la création d'objets graphiques, par exemple en DAO (dessin assisté par ordinateur) et CAO (conception assistée par ordinateur). Il faut donc envisager de créer des moyens de communication graphique avec des logiciels adaptés aux non-voyants.

C'est dans cette voie que certains chercheurs travaillent, en développant des tablettes graphiques et des souris Braille, où une matrice Braille éphémère est montée sur un chariot permettant des déplacements sur deux axes, affichant constamment en Braille les points qui se trouvent sous la tête. Le non-voyant peut alors en déplaçant la tête, dessiner des formes géométriques. Il existe un tel dispositif, développé à l'université de Stuttgart, formé d'une matrice d'interaction de 7000 points. Nous sommes bien sûr loin des 300 et quelques milliers de points d'un affichage à la norme VGA, mais nous sommes pourtant sur la bonne voie. D'autres dispositifs conçus à partir d'écrans tactiles et de reconnaissance de gestes sont à l'étude. Les systèmes d'exploitations multitâches (qui permettent d'effectuer plusieurs tâches quasiment en même temps), tels OS/2 et UNIX, sont bien évidemment les plus adaptés à de tels outils.

Citons d'autres travaux menés en Grande-Bretagne à l'université de York, concernant l'adaptation du Macintosh, où les informations de position d'une souris "ordinaire" sont transmises à l'aide de sons, couplés à une synthèse vocale. De même une interface permet de traduire de façon tactile une portion d'écran Macintosh (5 x 20 points -ou pixels-), et donc de pouvoir explorer l'ensemble d'un écran graphique. Le problème reste toutefois que la reconnaissance des objets (icônes...) est pourtant difficile avec de tels systèmes, et surtout très lente.

L'informatique dans la vie de tous les jours

Des efforts de recherches très importants ont été entrepris, en particulier au niveau de la CEE, afin de mettre

au point des dispositifs permettant une meilleure intégration du non-voyant dans son environnement, par exemple, en modifiant les distributeurs automatiques de billets à l'aide de synthèse vocale, ou en permettant la connexion de terminaux Braille types PC aux réseaux informatiques pour autoriser un meilleur accès des non-voyants aux services bancaires. Les terminaux vidéotex et en particulier le Minitel permettent dès à présent, qu'ils soient couplés ou non à un micro-ordinateur, un accès à tous les services bancaires, presse, Bourse, horaires de transports, réservations...

Le seul handicap à une utilisation massive de ces systèmes réside dans le coût actuel des cartes de synthèse vocale, et surtout dans les tarifs pratiqués pour ces services (abonnements, coût de connexion). Peut-être serait-il envisageable d'en modifier les conditions d'accès pour certaines catégories de personnes souffrant d'un handicap, la question est posée.
De même, dans le domaine de la Presse, l'informatique a permis des initiatives telles que celles prises par le quotidien régional Les Dernières Nouvelles d'Alsace. Les journaux étant saisis de plus en plus sur informatique, il est relativement aisé d'en produire une édition Braille. Ainsi, Les Dernières Nouvelles d'Alsace sont envoyées tous les jours aux abonnés non-voyants.

Mais le prix du papier est tel que vraisemblablement l'électronique et l'informatique prendront une part de plus en plus importante dans les services destinés aux non-voyants. Ainsi, des systèmes sont développés en Suède et au Japon afin de permettre un accès des non-voyants à la presse, via un système de radiodiffusion sur la bande FM, ou une transmission sur les lignes téléphoniques. Les informations sont émises la nuit, au cours du bouclage du journal, reçues par un micro-ordinateur, stockées, et automatiquement lues par synthèse vocale à la demande, affichées en Braille, ou en gros caractères sur un écran de télévision pour les malvoyants. Le coût de tels dispositifs peut paraître prohibitifs, mais qui pensait il y a encore peu, que le Minitel arriverait à détrôner l'annuaire papier ?
Bien évidemment, en raison de l'étroitesse de ce marché, la condition de la réussite de ces systèmes est vraisemblablement dans l'instauration d'une norme mondiale, permettant de diminuer les coûts de fabrication au maximum, et dans un effort des politiques sociales des états qui se sentent concernés.

> Le récent congrès cité dans l'article a donné lieu à la publication d'un ouvrage rassemblant l'ensemble des exposés, qui est disponible en noir et en Braille, ainsi que sous forme de disquette MS-DOS et Macintosh auprès de INFOVISIE MAGAZINE, Kapucijnenvoer 33, B-3000 Leuven, Belgique.

Et la musique dans tout cela ?

La musique reste le grand absent de l'innovation informatique au service des non-voyants. De nombreuses recherches ont été menées dans le monde entier, citons en particulier le Centre Tobia de l'université de Toulouse, le centre Marie-Morel de l'Institut National des Jeunes Aveugles, mais les partitions en Braille restent très difficiles à trouver. La traduction de musique en Braille est un problème très délicat, où il est urgent de voir apparaître des méthodes de traitement informatique, puisque les spécialistes en France de cette transcription se comptent sur les doigts de la main, et certains travaillent encore de façon manuelle.

Les principales voies de recherche pour permettre une diffusion massive de partitions Braille concernent la reconnaissance optique de partitions imprimées en noir, méthode encore naissante qui est promise à un bel avenir, et l'interfaçage d'un module de transcription sur un logiciel d'édition musicale du commerce, comme par exemple les travaux menés à l'université de Florence (Italie), et le logiciel en cours de développement chez une jeune maison française d'édition musicale, les Editions du Visage.

Gageons que les mois à venir permettront enfin de trouver, grâce aux outils informatiques, des partitions Braille dans les librairies musicales. Il est d'ailleurs à noter qu'une structure vient d'être créée en France, commune au ministère de la Culture et au secétariat d'État aux handicapés, qui a pour but de coordonner les opérations et informations relatives à l'accès des handicapés à la culture.

L'insertion dans le travail, clé de l'intégration des non-voyants dans la société

L'informatique, et sa généralisation dans l'entreprise, permettra de lever le principal obstacle relatif à l'intégration des non-voyants, en mettant à leur disposition un outil de communication par l'écrit qui faisait alors cruellement défaut. La technologie se doit d'être accessible à tous, et la société ne peut se permettre de restreindre les innovations qu'elle apporte à telle ou telle catégorie de personnes, au risque d'une désintégration de son tissu social.

La solution passe vraisemblablement par une formation et une qualification élevée du non-voyant, plus encore que pour le voyant. Comme le dit si justement N. Oriol dans son exposé au récent congrès consacré aux applications informatiques pour les aveugles : "... L'intégration d'un handicapé ne se limite pas à un coût économique provoqué par l'achat de matériel. Elle peut aussi s'analyser comme une sources de richesses..."

La généralisation des dispositifs informatiques adaptés aux déficients visuels ouvre un gigantesque espoir de réussite d'une intégration de tous les individus dans la société, qui est un impératif pour prétendre au titre de civilisation.

Denis Delbecq

> Une version Braille de cet article est disponible auprès de :
>
> Editions du Visage,
> Département Blind Concept.
> 7, rue Ccusté 94230 Cachan.



L'ART DE LA LIGNE A LA PECHE !

Dans une gamme de prix sensiblement équivalente (moins de 3000 F), Outline Art voit arriver sur le marché un concurrent des plus sérieux, distribué par ALM, probablement disponible au moment où vous lirez ces lignes, en la personne de Didot LineArt.

L'environnement logiciel de l'utilisateur de PAO semble devoir s'étoffer très sérieusement en ce début d'année. Tous les outils professionnels dont il peut avoir besoin vont être (ou sont déjà) disponibles sur sa machine, qu'elle soit ST ou TT. A l'annonce de l'arrivée imminente du déjà célèbre, tant il nous fait languir, Calamus SL, on peut ajouter celle de Didot LineArt ainsi que celle de Retouche Pro (que nous testerons dès le prochain numéro). De quoi combler toutes les attentes !

DE DIDOT LINEART DONC...

Ce programme se présente comme une version largement étendue de Didot Fonteditor, sorti l'année dernière (une mise à jour est prévue pendant 6 semaines pour une somme représentant la différence). La partie éditeur de fontes est conservée et améliorée, y est adjointe une partie graphisme vectoriel qui n'est pas sans rappeler, tout en s'en démarquant de par ses nombreuses possibilités, Outline Art.
Il fonctionne avec 1 Mo minimum de mémoire, en mode haute résolution monochrome ou sur grand écran en ce qui concerne les ST, ainsi que dans les modes du ST sur le TT. Il s'installe sans problème sur disque dur ; mais avant cela, un petit programme d'installation inscrira le nom de l'utilisateur de manière indélébile dans le programme principal : ceci assurera le suivi quant aux mises à jour.
Au lancement du programme, on arrive sur une page dont la partie gauche est occupée par des icônes. Il s'agit de la page de menu principal, sorte de gare de triage à partir de laquelle on pourra se rendre dans les différentes parties du programme, ou encore accéder aux périphériques de sauvegarde ou d'impression. Une icône mène à la partie dessin bitmap, mais en réalité, il ne s'agit dans cette partie que de pouvoir charger une image de ce type en tant qu'image de fond ou image à vectoriser.

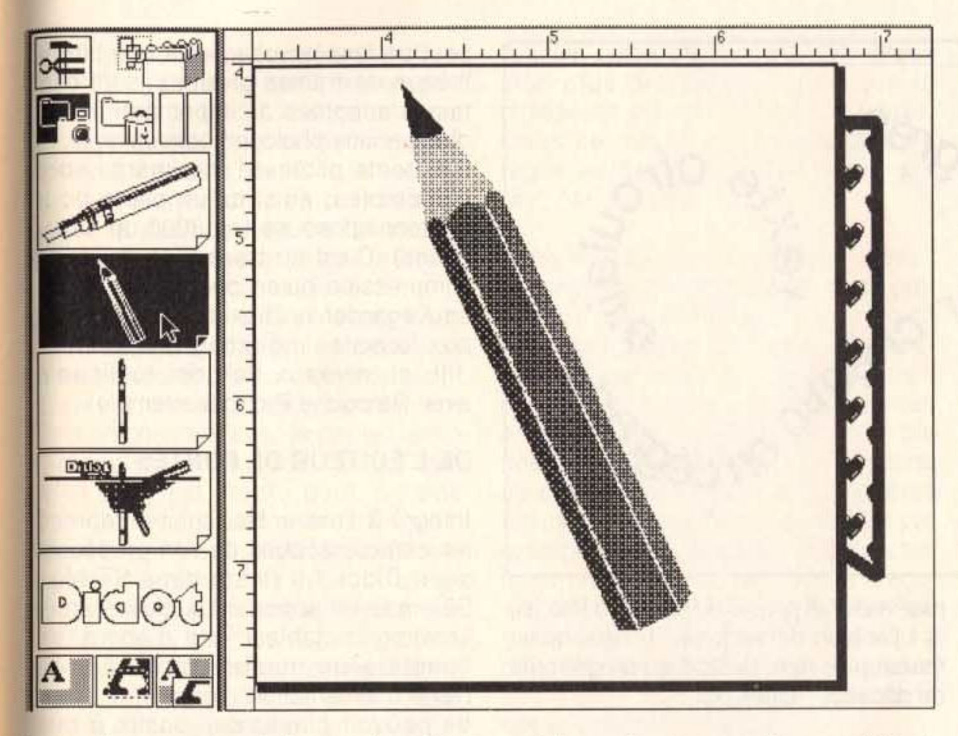
DU CONFORT TOUT PARTOUT...

Après avoir cliqué sur l'icône qui permet d'y accéder, on arrive au coeur du programme. La présentation reste la même, à savoir un écran de dessin bordé sur la gauche d'icônes, différentes selon le mode de travail.

La hiérarchie du programme est la suivante : un objet est constitué d'un ou plusieurs vecteurs, eux-mêmes constitués de droites et/ou de courbes (de Bézier), elles-mêmes constituées de points de contrôle simples, ou tangentiels dans le cas des courbes. Le travail de base consistera donc à placer, déplacer, ôter et insérer des points aux endroits judicieusement choisis, afin de créer des graphismes à vous couper le souffle (si, si !).
L'écran de travail peut être bordé de réglettes de mesure, le format de la page ayant été auparavant librement défini. Les coordonnées de l'outil courant sont continuellement visibles à l'écran, de manière absolue et relative.
Un zoom très puissant est disponible, autorisant un travail extrêmement précis. On peut de plus rappeler l'ancien facteur de zoom et passer ainsi très rapidement de l'un à l'autre. Un clic sur le bouton droit permet enfin de placer la fenêtre virtuelle de travail à l'endroit que l'on désire sur la page.
Une aide en ligne rappelle sur la ligne supérieure de l'écran la signification de l'icône sur laquelle se trouve le curseur. L'aide pousse le raffinement jusqu'à indiquer les raccourcis clavier ainsi que les éventuelles possibilités de "double clic". De la sorte, certaines icônes donnent accès à un sous-menu qui autorise différentes options. Enfin, pour couper court aux perpétuels râleurs qui contesteraient le choix de tel ou tel raccourci clavier, il est prévu de les modifier à loisir.

DU DESSIN VECTORIEL...

Il y a en fait trois modes de dessin possibles : la création d'objets, la manipulation d'objets et la création/manipulation d'objets de type texte.
Le premier de ces modes est donc à la base de tout dessin. Pour ce faire, il s'agira de placer des points commandant des droites ou des courbes. Ces points sont déplaçables à volonté. Il est également possible d'en ajouter (insertion), d'en supprimer, d'en raccorder, de modifier le type de ligne. Une option autorise de plus le dessin à main levée, ce qui se traduit, une fois cette opération terminée, par une succession de droites suivant le tracé effectué. De très nombreuses possibilités d'aimantation, détaillées dans la partie éditeur de fontes de cet article, agrémentent le travail d'une précision accrue.
Des "clipboards" (mémoires intermédiaires) permettent de stocker des vecteurs, d'en assurer la copie ; ceux-ci ne sont limités que par la place mémoire disponible. Leur mode de visualisation en a été amélioré (voir aussi la partie éditeur de fontes).



Lorsqu'on en a terminé avec la création d'un objet, un clic sur l'icône appropriée donne accès aux outils de manipulation d'objets, le second mode de travail. Les possibilités y sont très nombreuses, de par un grand nombre d'options diverses et de modalités de travail bien pensées.

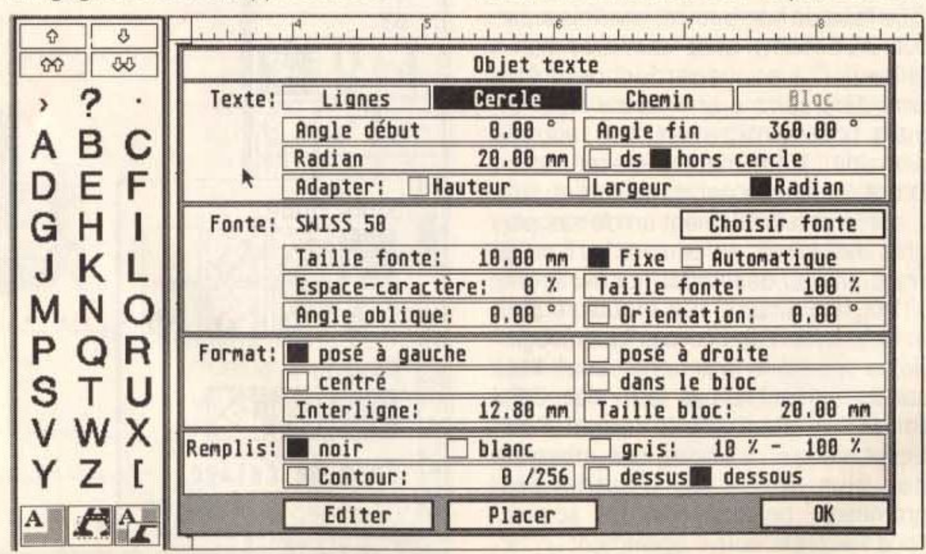
DE LA MANIPULATION D'OBJETS

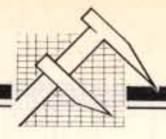
Les deux premières options, inhabituelles et pourtant ô combien utiles, permettent de déterminer pour chaque objet l'épaisseur de la ligne, son type (continu, tirets divers), son motif ou niveau de gris, ainsi que le mode de remplissage de la surface en niveaux de gris ou par un motif de type GEM.
Chaque objet peut subir des déformations horizontales ou verticales, proportionnelles, être incliné horizontalement ou verticalement (y compris en indiquant l'angle exact), subir une rotation (ici aussi en indiquant l'angle). Ces différentes transformations se font quasiment en temps réel, en saisissant un angle du cadre entourant l'objet ou encore en "attrapant" un des points de contrôle des vecteurs constituant l'objet. Ces nombreuses possibilités autorisent des "triturages" dans une gamme vraiment très large.
Un des avantages des objets vectoriels est qu'ils peuvent être empilés. Cela représente pourtant parfois une perte de temps, lorsqu'il s'agit de modifier l'objet situé justement, comme un fait exprès, tout en dessous. Il faut aller le chercher, en envoyant bien souvent les objets qui le couvrent en dessous, ou en les déplaçant de manière à le faire revenir "à la surface", puis en remettant tout en place. Rien de tout cela avec Didot LineArt ! Les touches shift, control et alternate judicieusement utilisées permettent de sélectionner l'objet que l'on veut, quelle que soit sa place dans un empilement de cette sorte. Mieux, il est possible d'envoyer un objet à l'endroit voulu de l'empilement, à savoir au-dessus ou en dessous (classique), mais également aux positions intermédiaires (moderne !). Inutile d'ajouter que le temps et la commodité de travail gagnés sont fort appréciables.

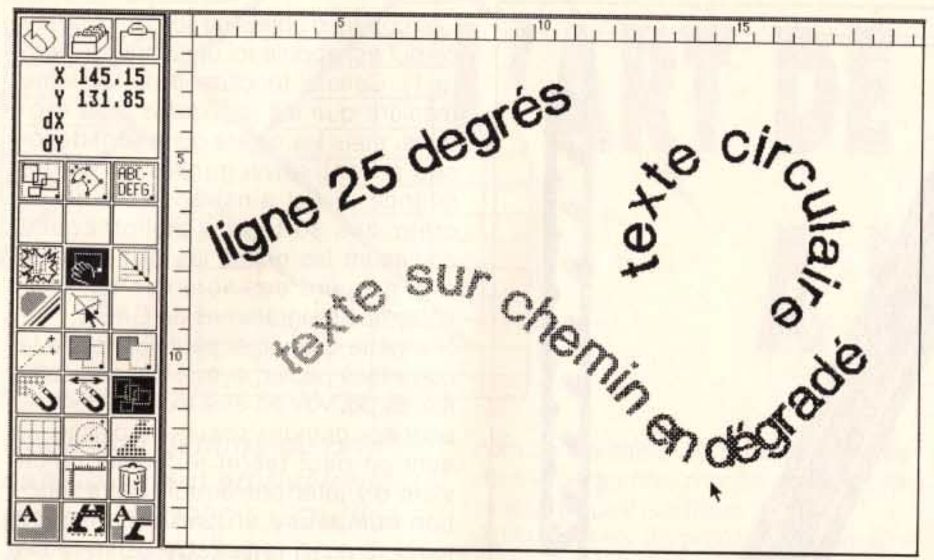
Les copies d'objets se font à l'aide de ce qui est appelé ici une librairie (figure 1). Celle-ci fonctionne de la même manière que les clipboards déjà évoqués, mais les objets qui y sont déposés seront sauvegardés en fin de séance. Il est ainsi possible de se créer des sortes de bibliothèques contenant les objets les plus utilisés, à la manière des librairies de symboles des programmes de CAO.
N'importe quel objet peut être jeté à la corbeille à papier, avec cette particularité de pouvoir en être extrait : comme pour ses grandes soeurs en plastique, dont on peut retirer le feuillet qu'on vient d'y jeter par erreur. Cette fonction compense en partie l'absence toujours aussi regrettable d'une réelle fonction UNDO. On pourra également, pour poursuivre l'analogie, "vider" la corbeille dans le vide-ordures et donc en perdre définitivement le contenu.

DE L'OBJET TEXTE

Une icône fera apparaître un formulaire permettant de régler les nombreux paramètres afférents au texte (figure 2). Tout d'abord, en déterminer le type : texte sur une ligne (256 caractères maximum) d'inclinaison variable, texte sur ou dans un cercle avec angle de départ et d'arrivée, ou enfin texte sur un "chemin". Dans ce cas, il s'agira au préalable de dessiner à l'aide de vecteurs (droite et/ou courbe) le tracé que devra suivre le texte (figure 3).
Ce texte s'écrit à l'aide des polices de type Calamus, dont on peut charger plusieurs exemplaires à la fois.
Les autres réglages concernent la taille de la fonte, son espacement,







son angle d'italique, son angle de rotation, son degré d'étroitisation ou d'élargissement, sa "couleur", l'épaisseur de son trait de contour, ou enfin sa valeur de dégradé. Ouf !

L'objet texte, une fois placé, peut subir les transformations déjà citées, à l'exception de la modification du contour ou de la surface.

Si tous ces paramètres ne permettent toujours pas d'arriver à l'effet désiré, il est possible de "dégrouper" l'objet texte. Cela revient à dissocier chaque caractère, qui sera alors considéré comme un objet vectoriel simple. Cette opération fait perdre les caractéristiques proprement textuelles, mais permet maintenant toutes les manipulations. Une dernière précision : regrouper ensuite les caractères ne reforme pas un objet texte.

DE L'OUVERTURE

Que faire de son oeuvre, une fois achevée ? L'exporter, avez-vous répondu en choeur ("la sauvegarder" peut être considéré comme une bonne réponse, mais l'on devra revenir en seconde semaine...). En matière d'exportation, Didot LineArt n'est donc pas avare. C'est incontestablement un de ses plus gros avantages, en ce sens qu'il ne se limite pas au désormais familier format .CVG (Calamus). Il y ajoute le sien propre, avec dans ce cas la sauvegarde de la session entière de travail. Mais surtout, il n'oublie pas le format .GEM (metafile), ainsi que le format PostScript, ce qui lui assure une ouverture des plus larges sur d'autres programmes : un régal pour les applications PAO. On trouve également un format vectoriel propre à Retouche Pro (si, si ! j'ai bien dit vectoriel...), ainsi qu'un format plus rare, destiné au programme de découpe : GMA.plot.

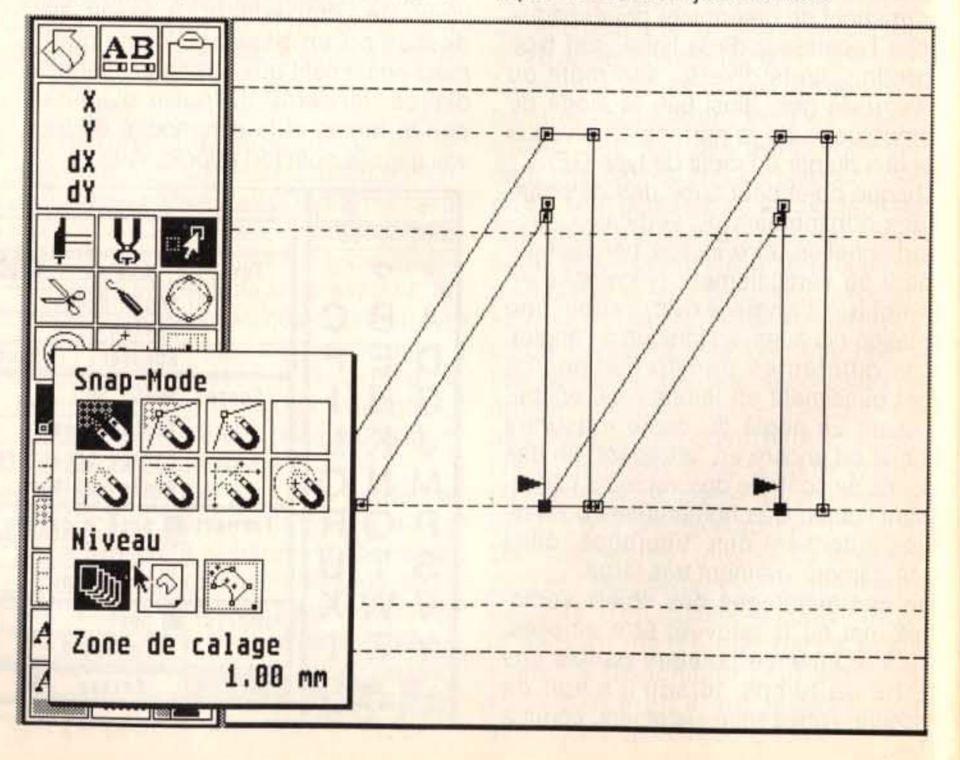
DE L'IMPRESSION

À la question du paragraphe précédent, on pouvait également répondre : "l'imprimer". Pas de problème, c'est prévu. Le format de la page est paramétrable, ainsi que le nombre d'exemplaires, même l'impression en négatif est possible. On pourra adapter la taille du graphisme au format de la page, ou encore choisir d'imprimer la page entière ou bien un (des) objet(s) sélectionné(s). Le choix de la trame peut se faire en chargeant une bibliothèque de trames diverses, dont certaines adaptées à l'imprimante laser, d'autres aux photocomposeuses.

Différents pilotes d'imprimante sont disponibles, ainsi qu'un pilote pour photocomposeuse (en 3000 dpi théoriques). C'est au travers de ce menu d'impression qu'on pourra également sauvegarder en fichier le graphisme aux formats .img (images bitmap) et .tiff, en niveaux de gris, (utilisable avec Retouche Pro, par exemple).

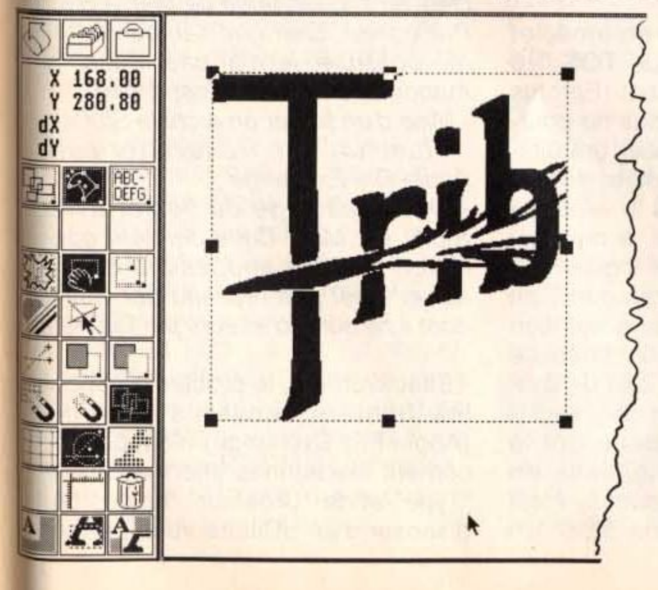
DE L'ÉDITEUR DE FONTES

Intégré à l'ensemble, celui-ci reprend les caractéristiques de son prédécesseur, Didot 1.4 (testé dans ST Mag 36), mais en apportant quelques améliorations notables. Tout d'abord, on constate un nombre croissant de types d'aimantation (figure 4). Il s'agit de pouvoir placer des points à des positions rendues "magnétiques" au préalable, positions qui attirent de la sorte le curseur. On a le choix entre "magnétiser" une grille entièrement paramétrable, des lignes d'aide verticales et/ou horizontales, les points déjà placés, des cercles d'aide, le tout affichable à volonté. La précision s'en trouve fortement augmentée, et placer dans ces conditions les quatre points nécessaires à la création d'un "o" à l'aide d'un cercle d'aide "magnétique" devient un jeu d'enfant.



Chaque courbe se contrôle par deux points tangentiels dont la position déterminera le tracé. Il est dorénavant possible de maintenir ou non les deux points tangentiels liés. S'ils le sont, le déplacement d'un des points provoque le déplacement de l'autre, ce qui permet d'obtenir (entre autres) des effets d'arrondi plus lissé.
Les "clipboards", déjà présents dans la précédente version, se sont eux aussi améliorés. A leur nombre théoriquement illimité, il faut ajouter leur mise en commun avec la partie dessin vectoriel du programme. Enfin, tout objet qui y est stocké peut, à l'aide d'un clic, être visualisé en taille plein écran avant d'être replacé sur la surface de travail, si on le désire. Cela se révèle particulièrement utile pour distinguer deux objets d'aspect proche. Ce type de visualisation existe maintenant également pour ce qu'on "jette" dans la corbeille à papier.
Une des grandes nouveautés est la possibilité de régler l'approche des caractères par paire. Rappelons ici que l'approche est le blanc nécessaire entre deux lettres imprimées d'un même mot pour qu'elles ne se touchent pas, alors que la chasse est la largeur du caractère (de l'oeil) y compris les approches. L'ancien mode de réglage de l'approche, manuel ou automatique, est conservé, mais y est donc maintenant ajouté celui du réglage par paire, ces modes étant incompatibles entre eux. Ce nouveau mode, facilité par la présence de barres verticales servant à déterminer la chasse de chaque caractère, permet de régler l'approche de ceux-ci deux par deux (par paire), au choix de l'utilisateur. A l'usage, cette fonction offre une précision plus grande pour peu que la chasse ait été correctement indiquée. Dans ce cas, il ne reste plus qu'à régler les "paires cruciales" telles que "av", "va", "li", etc.

Autre nouveauté et non des moindres : la possibilité de charger une fonte protégée, c'est-à-dire portant le numéro de série du Calamus sur lequel elle fonctionne. Quel intérêt ? Tout simplement de pouvoir, à partir d'une telle police, en créer une nouvelle, par exemple une déclinaison italique. Compte tenu du prix de ces fontes professionnelles (telles les célèbres Compugraphics par exemple), cette possibilité est franchement bienvenue et génératrice d'économies. La nouvelle fonte ainsi créée gardera le numéro de série et ne sera donc pas diffusable. On peut supposer que c'est à cette condition que cette option a pu être intégrée.
Une autre façon d'agrandir son stock de polices de caractères est proposée. On peut, en effet, charger des polices vectorielles de type PostScript et les sauvegarder au format propre à Calamus. N'ayant pas de telles polices à disposition, nous n'avons pu effectuer de test pour en vérifier l'efficacité, le rendu ou la fiabilité.
Pour en terminer avec l'éditeur, je citerai encore ce confort nouveau qu'a l'utilisateur en se rendant au besoin dans la partie dessin vectoriel afin d'y visualiser ses caractères en situation réelle, à savoir dans du texte, puis de revenir effectuer les éventuelles retouches nécessaires. Ceci apparaît très vite indispensable, particulièrement lors du réglage de l'approche par paire. De même, pouvoir imprimer sans avoir à passer par une sauvegarde, puis le chargement de Calamus, puis l'importation de la police à tester représente un gain de temps appréciable.
On est cependant amené à regretter certaines absences, telles que la possibilité d'étroitiser ou d'incliner l'ensemble de la fonte en une seule opération. De même la gestion d'une image de fond servant de modèle semble moins efficace que dans la version antérieure. La version "pro" comblera peut-être ces attentes, comme elle offrira peut-être, rêvons un peu, la vectorisation en courbes de Bézier, le tout directement dans l'éditeur.
Incontestablement, cette nouvelle mouture offre un progrès marqué avec ce qu'elle apporte de confort, de précision et d'agrément supplémentaires dans le travail parfois ingrat que représente la création ou la modification d'une police. Il est évident que ses concepteurs ont eu présent à l'esprit la volonté de faciliter ce type de travail.



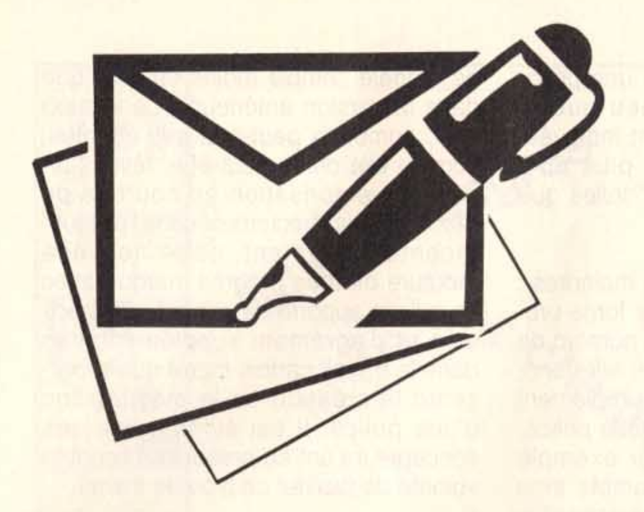
DU MODULE DE VECTORISATION

Livré avec le programme, ce module externe ne se chargera qu'en cas de besoin. Il faudra au préalable importer une image de type bitmap ; on encadrera ensuite la portion d'image à vectoriser, et le logiciel se chargera du reste : une image de grande taille pourra être vectorisée en plusieurs fois. La vectorisation ne se fait qu'avec des droites, mais les résultats en sont d'assez bonne facture (figure 5). Notons toutefois que son utilisation amène parfois à planter le programme, ce qui demeure regrettable.

DE LA CONCLUSION

Si l'on excepte la "calculette" et la projection sur une surface, ce programme se révèle largement supérieur au seul qui existait jusqu'à présent pour ce type d'application, à savoir Outline Art. Le nombre de possibilités, la puissance, le confort de travail en font un outil d'une qualité remarquable. Si l'on ajoute à cela son esprit d'ouverture qui ne le limite pas au seul Calamus, son aspect professionnel (liaison avec Retouche Pro, photocomposition, programme de découpe...), on peut affirmer avoir affaire à un logiciel adulte. Finalement, peu (mais tellement beaucoup !) de choses le séparent d'un programme de PAO. Il est tout à fait possible de réaliser, par exemple, des documents d'une page ne comportant pas trop de texte, où l'aspect graphique domine, de manière totalement autonome. ALM annonce une version "pro" à venir (date indéterminée), mais il semble qu'on aurait tort d'attendre, tant cette version-ci se révèle déjà aboutie.

Patrick Bonnet



COURRIER DES LECTEURS

Pourriez-vous me donner une réponse au problème suivant ? Je suis muté à Tahiti (ça, ce n'est pas un problème), et j'emporte mon matériel informatique, plus un synthétiseur. Or, le courant électrique à Tahiti est de 220V / 60 Hz... Voilà le problème !

- Aucune solution miracle à vos soucis, à part le changement systématique de toutes les alimentations présentes dans chacun de vos appareils (coût + délais), ou l'utilisation non moins coûteuse d'un onduleur (5 à 6000 F).

Equipé d'un Amiga 2000 et de matériel vidéo (essentiellement S-VHS), je me suis procuré, suite à l'article paru dans votre numéro 46, le GST 2500 de la société SATV, dans le double but de disposer d'un genlock de qualité pro et de pouvoir réaliser des incrustations en "Chroma Key". Or, je me retrouve dans l'impossibilité de d'effectuer correctement mes incrustations : dans le cas d'une silhouette devant être découpée sur fond bleu, il reste un liseré de bleu assez épais d'un côté, la silhouette se trouvant de l'autre amputée de la même épaisseur; la découpe de l'image est donc rendue inutilisable par ce décalage horizontal. J'ai depuis quelque temps fait part de mes soucis à la société SATV, où j'obtiens invariablement cette réponse : "nous allons bientôt vous fournir un retardateur". Y-aurait-il moyen, a votre connaisance, d'entrevoir une réponse à ce problème ?
Mr Schutz, Courbevoie.

- Le "retardateur" constitue en effet le complément nécessaire aux installations vidéo qui ne sont pas asservies sous un seul et même signal "Blackburst" (lequel regroupe les signaux de synchronisation), ou lors de traitements de signaux occasionnant un retard image (cas des "Chroma Key"). Dans le but de répondre à une large palette de problèmes de cet ordre, la société SATV vient, renseignements pris, de terminer la mise au point d'un retardateur variable, qui pourra être calibré par l'utilisateur en fonction de ses besoins. Sa commercialisation est prévue pour le début de cette année.

Je suis confronté à un "Bug" malheureux du Tos 1.6 : ce dernier refuse de sauvegarder sur disquette la configuration du bureau en moyenne résolution. Malgré les différentes "bidouilles" désormais connues, n'existe-t-il pas une solution "radicale" pour ne pas avoir à modifier systématiquement le Desktop.Inf ?
Armand Gérard. 35 LIFFRE.

- Oui, c'est un vrai problème, et vous devez savoir qu'un TOS 1.6 corrigé existe officiellement (Eproms disponibles) mais que vous ne pourrez vous le faire remplacer gratuitement. En effet, les garanties reconnues juridiquement dans la vente en informatique portent sur le matériel (hardware) et non sur le logiciel (on trouve toujours, généralement en tous petits caractères, une mention signifiant que la responsabilité ne saurait être engagée en cas de dysfonctionnement du logiciel, etc. - c'est d'ailleurs le cas pour tout le secteur de l'édition logicielle en général). Donc, pour résumer, il est possible d'ACHETER un TOS 1.6 corrigé, le seul conseil possible en la matière étant, si possible, d'attendre les environs du Printemps prochain, car un TOS 2.0 en français, bénéficiant des progrès réalisés avec le TOS du TT, sera également diffusé par Atari. Quitte à acheter un nouveau jeu d'Eproms, autant se procurer celui-là...

- Existe-il un livre traitant des fonctions du GfA Basic suivantes : Defmouse/Fill.
R.Botton. Versailles.

- Tout simplement : le livre du GfA Basic 3.0, chez Micro applications.

A propos des transferts de fichiers entre Mac et Spectre, que nous abordions en particulier dans le numéro de mars dernier, Robert Pellerin nous fait part de ses diverses remarques, et nous dit:

"Je n'ai personnellement connu aucun échec pour transférer logiciels ou fichiers texte de Mac à ST. Par contre, bien que l'auteur l'affirme possible, je n'ai pas réussi la manoeuvre inverse, c'est-à-dire :
- Mise d'un fichier en archive Stuffit ;
- Transfert par Transverter puis Apple File Exchange ;
- Décompactage du fichier avec Stuffit sur Mac. Cette dernière opération s'avère impossible car le fichier n'est pas reconnu par Stuffit dont il ne porte d'ailleurs par l'icône."

- Effectivement, le problème est que les fichiers convertis par AFE (Apple File Exchange) n'ont pas forcément les bonnes informations de "Type" et de "Créateur". Il vous faut disposer d'un utilitaire vous permet-

tant de modifier ces informations (la plupart des utilitaires de disque, ou encore ResEdit), pour leur attribuer des valeurs appropriées, c'est à dire "SIT!" pour tous deux (les 3 lettres SIT et un point l'exclamation). Transverter a cet avantage sur AFE qu'il permet de définir ces attributs lors du transfert.

Pour sa part, Eric Monseau nous parle de ses inquiétudes au sujet de son disque dur, un Megafile 30 :

"Il marche très bien mais j'ai des temps d'écriture vraiment longs, ce qui commence à me taper sur les nerfs !
Lors de l'écriture il commence par une vingtaine de petits accès, puis termine par un gros. La copie d'un fichier de 10 Ko dure 7 à 10 secondes de RAM-disque vers disque dur ! Autant dire qu'il faut éviter de copier 100 fichiers de 1 Ko (cela prend 30 mn !). Là je me suis dit qu'il y avait vraiment un problème. En tous cas il est moins critique lors de l'écriture de gros fichiers, ce qui est normal car le problème semble être la création de fichier. Les lectures sont, elles, correctes pour un disque de cette vitesse.
Ma configuration est la suivante: 1 Mo de RAM, TOS 1.0 (86), le disque dur est partitionné en 2, 14 et 16 Mo pour C, D, E respectivement. Le driver est AHDI 3.01. Il est à noter que la partition C est la plus rapide des trois. Il ne s'agit pas aussi de virus, ce problème étant survenu dès l'achat.
Mon revendeur avoue être perplexe devant mon cas. Je pense qu'il s'agit d'un problème de logiciel, soit de driver, soit de TOS (passer au TOS 1.2 ?)."

- Voilà un problème qui n'en est malheureusement pas un. La gestion des disques étant ce qu'elle est dans le TOS (avant le 1.4), vous êtes astreint à supporter cette lenteur, à moins de vous livrer à l'une des manoeuvres suivantes :
- Passer au TOS 1.4. Il n'est absolument pas officiel, et ne le sera semble-t-il pas avant un bon moment, mais si vous pouvez vous procurer les EPROMs correspondantes (ça "circule" beaucoup), n'hésitez pas, c'est radical.
- Utiliser TurboDos. C'est un programme écrit par Dominique Laurent, développeur controversé du monde ST (Emulcom, Le Rédacteur...), qui utilise un principe similaire à celui du TOS 1.4, c'est-à-dire l'utilisation de tampons pour les secteurs de FAT, de répertoires, et quelques autres.
Attention, comme je vous le disais, Dominique Laurent est très controversé, et TurboDos, comme le Rédacteur, est considéré par une partie des utilisateurs comme un programme merveilleux et n'ont jamais rencontré le moindre problème en l'utilisant, l'autre moitié clamant avoir perdu de nombreuses heures de travail par sa faute. Faites ce que vous voulez, je ne ferai que citer un programmeur d'Atari Corp : "TurboDos, that buggy french thing".
Personnellement, j'ai opté pour le TOS 1.4, au détriment de la légalité, TurboDos étant lui totalement gratuit (dans le domaine public).
Vous ajoutez dans votre lettre une foule de compliments dont nous vous remercions, en ajoutant la remarque : "elle est bien équilibrée (quoique un peu trop de Basic à mon goût et pas assez de programmes C)". Je suis tout à fait complètement totalement irrémédiablement d'accord. Seulement, une bonne partie des programmeurs C de l'équipe utilisent Turbo C, et celui-ci n'étant pas disponible en France, nous avons longtemps attendu qu'il le soit avant de vous livrer des dizaines de pages l'utilisant, mais ceci risque de changer, on commence à être fatigués d'attendre, et le meilleur compilateur C sur ST vaut bien qu'on se fatigue à l'acheter en RFA, non ?

De Suisse nous parvient une lettre de Stéphane Pellissier, qui aborde un sujet bien intéressant, puisqu'il s'agit du TT (rêve, rêve...) :

"Les premiers TT commencent à arriver chez mon revendeur, et l'excitation me gagne chaque jour un peu plus. Devant l'incapacité de ce charmant personnage à me renseigner, je prends mon traitement de texte préféré, et me résous à vous soumettre mes angoisses :
1. Pour répartir quelque peu les dépenses, j'envisage d'acheter très rapidement un moniteur multisync (Eizo 9070-Z/16" ou Nec 4d/16"). Je crois savoir qu'un tel moniteur se branche parfaitement sur mon Mega ST. J'aimerais savoir si cela est aussi le cas pour le TT, et en particulier, pour le mode haute résolution 1280x960. En outre, un tel moniteur supportera-t-il la future extension graphique, la carte "true color" qui équipe l'ATW 800 (dixit Sam Tramiel) ?
2. Est-ce que la version TT/TOS pourra facilement évoluer vers la future version TT/Unix. Si oui, quelle est l'importance des modifications à effectuer, et sera-t-il possible de conserver simultanément les deux environnements ?
3. Qu'en est-il de l'annonce d'Atari, de fournir avec le TT un émulateur MS/DOS, développé par Insigna et surtout quel standard du monde PC est-il capable d'émuler (8088, 286, 386SX, ou plus) ?
4. Est-il plus intéressant d'investir dans la gamme de disques durs actuels (Megafile ou autres) prévus pour le ST et son port DMA ou de patienter un peu, et de se tourner vers le grand choix de disques durs au standard SCSI, pour offrir un disque dur extractible à mon ordinateur ?
5. Enfin, est-il possible de connecter le TT, sans bidouilles, à un lecteur HD 1.44 Mo, compatible IBM et Mac, en interne, ou en externe ?"

Pffffiou. Que de questions d'un coup, mais quelles questions intéressantes ! Allons-y, attaquons.
Pour les moniteurs Multisync, il n'y a effectivement pas grande difficulté à en connecter sur un ST ou STE, et il devrait en être de même sur TT, pour les 5 premiers modes graphiques (ceux du ST, 320x480 et 640x480). Notez bien que le TT, contrairement au ST, travaille toujours à la même fréquence dans tous ces modes, et ne requiert donc pas un moniteur multisync pour profiter de toutes.

Le mode haute résolution, quant à lui, devrait poser quelques difficultés. Je ne pense pas en effet que les moniteurs que vous citez supportent des résolutions aussi élevées, mais il suffit normalement de consulter les caractéristiques des dits moniteurs pour s'en assurer.



Pour finir avec les écrans, la carte graphique étendue, dérivée de celle de l'ATW, n'a pour le moment pas encore été présentée, il est donc difficile de se prononcer. Si on se fie aux caractéristiques de l'ATW (ouuuhhh que c'est loin), il y a plusieurs modes graphiques, les plus faibles (et plus colorés) devant passer sur les moniteurs sus-cités, les plus élevés (il me semble qu'on a aussi ici du 1280x960, mais en couleur en plus) devant avoir plus de mal...

Il n'y a strictement aucune différence entre un TT sous TOS et un TT sous Unix ! Rappelons qu'il n'existe pour le moment qu'une seule machine, le TT (de bureau, disons), qui tourne sous TOS, et pourra tourner sous Unix par simple installation de celui-ci (du logiciel) et quand même d'un disque dur approprié (48 Mo pour Unix, c'est même pas la peine d'y penser). Comme nous le disions dans le dernier numéro, Atari France pense que l'update vers Unix se fera par simple échange du disque dur, et que le coût de l'opération devrait être sensiblement identique au prix normal du disque dur vide. Le TOS sera toujours disponible (les ROMs resteront), mais je doute de la possibilité d'utiliser des applications TOS tout en laissant tourner Unix, ce qui risque de rendre l'opération assez malaisée.

L'autre machine annoncée, le TT/X sera techniquement identique au TT (aux dernières nouvelles, en tous cas), et ne se distinguera que par son boîtier de tour, un plus grand nombre de slots VME (qui seront en A32:D32 en plus), et quelques détails "mécaniques" de cet accabit.

Pour l'émulateur PC d'Insigna, qui porte le doux nom de "Soft PC", et qui est un portage du logiciel de même nom sur Mac II, il s'agit, a priori, d'un émulateur 286, et nous ignorons les derniers détails concernant sa fourniture ou non.

Passons aux disques durs amovibles. La technologie la plus répandue à l'heure actuelle est celle de SyQuest, et les mécaniques de ce constructeur (lecteur SQ555, à la norme SCSI) sont au coeur de tous les lecteurs de ce type actuellement vendus, y compris le MegaFile 44. Les versions pour ST étant simplement dotées en sus d'une carte DMA-SCSI permettant leur connexion à un ST. Donc, quoi qu'il arrive, vous achetez toujours la même chose, mais vous ne payez pas forcément le même prix pour ce qu'il y a autour.

Quant aux lecteurs haute densité, le TT aurait dû en disposer en standard, mais, malheur, grand malheur, ce n'est pas le cas. Il n'est pas possible de connecter directement un lecteur de ce type (ou alors pour l'utiliser en mode 720K habituel) à moins d'effectuer quelques modifications à l'intérieur du TT (qui seraient déjà prévues, on enquête...).

De son côté, F. Colomb nous écrit :
"Je travaille actuellement sur un 1040STF, associé à un Megafile 30 et une LC24-10. Je souhaiterais disposer d'un second poste de travail pour un coût réduit, d'où ma question : est-il possible d'utiliser "à deux" (soit simultanément, soit alternativement) le disque dur et l'imprimante ? Quelles sont les contraintes techniques spécifiques pour la connexion du disque (longueur des câbles, etc.) ?"

- Voilà un problème courant que celui du partage des ressources. Le cas de l'imprimante peut être résolu de façon simple par l'emploi d'un boîtier de connexion d'imprimante "2 entrées, 1 sortie", qu'on trouve chez d'assez nombreux revendeurs (au hasard, citons Komelec, qui est spécialisé dans la connectique). Sur ce boîtier, vous connectez les 2 postes et l'imprimante, et un simple bouton vous permet de choisir le poste qui dispose de l'imprimante.
En ce qui concerne le disque dur, c'est plus délicat. Une solution consisterait à employer un réseau local, il en existe de performances et de prix variés, allant du réseau par prise MIDI, relativement lent mais pas cher du tout (il n'y a pratiquement pas de matériel supplémentaire) au réseau Ethernet, rapide (plus d'un méga-octet à la seconde !) mais cher (plus de 5000 francs par poste, donc inintéressant pour le partage d'un disque dur à 4000 francs !), en passant par le réseau Arcnet, de performance et de prix moyens (mais encore une fois, l'investissement pour 2 postes sera supérieur au prix d'un nouveau disque dur !).
Franchement, un deuxième disque dur paraît être la meilleure solution, à moins d'avoir à partager des données. Dans ce cas, il vous faudra déterminer le volume de données à partager, et choisir un réseau en fonction de ce critère (et du budget accordé à cet investissement).
Un dernier détail : la société Eickmann, en Allemagne, propose une certain nombre d'interfaces spéciales pour disque dur, dont un système permettant la connexion de plusieurs postes à un même disque dur. Mais ce dispositif n'étant pas disponible en France...

En ce qui concerne la connexion du disque dur, la seule remarque que l'on puisse faire est que le DMA ne supporte pas énormément les câbles un peu longs (à cause de son haut débit, et de l'absence de protocole de contrôle).



NEWS

CIS NIOUZES

Nous vous avions déjà annoncé l'existence de Broadcast Titler 2. CIS, l'importateur officiel, propose dès maintenant la version française du logiciel et du manuel. De plus, certains bugs mineurs ont été revus par rapport à la version anglaise.
Cette version n'arrive pas seule, elle est accompagnée du nouveau "font enhancer" qui permet de convertir les fontes Amiga en fontes BT2 "anti-aliasées". Cet utilitaire permet aussi de réduire au format voulu ces fontes, et de créer des jeux de fontes pour le nouveau module 3000. Mais voici la vraie nouveauté de BT2 : un module destiné aux possesseurs d'A3000 permet d'utiliser les nouveaux modes graphiques. Vous pourrez composer vos pages avec une résolution de 1270x560 pixels (mode super hires), et utiliser un maximum de 4 couleurs par ligne, pour un total de 64 couleurs par pages. Il est important de savoir que les fontes restent spécifiques, et qu'il vous faudra posséder un A3000 avec une révision 36202 du Kickstart et 3677 WB pour en profiter. Les premiers fonts-packs arrivent eux aussi, de quoi convaincre les plus perplexes.

BROADCAST
TITLER 2

Texturesinfinies

effetsspeciaux

BT2 sur A2000 : mieux qu un long discours...

CIS propose la dernière version de DigiPaint3, sans oublier le manuel, tout en français. Elle a été modifiée pour tourner complètement sous A3000, son prix conseillé : moins de 800 F.

G.V.P. propose une nouvelle carte d'extension mémoire en barrettes SIM pour A2000. Elle est disponible en version 2, 4, 6 et 8 Mo pour un prix s'échelonnant, par tranche de 1000 F, de moins de 2000 F pour la 2 Mo à un peu moins de 5000 F pour la 8 Mo. La version 6 Mo est enfin une solution pour les utilisateurs de carte passerelle PC.

M.M.U. 1.0 : Music Manager Unit sera dès le début février vendu pour moins de 800 F avec PerfectSound 3.0. Il fonctionne dès à présent sous le WB 2.0.



Choix des effets possibles sur chaque ligne

Choix des effets de transitions entre chaque page

Un nouveau disque dur à cartouche amovible, le RH 5500, est soutenu par CIS. Il utilise des cartouches de 50 Mo, non compatible Syquest. Il vous en coûtera 9500 F pour l'unité, la cartouche valant un peu moins de 1400 F. L'écart de prix est justifié par une nouvelle technologie augmentant nettement la fiabilité. Un service de conversion de cartouche est mis en place pour assurer une continuité à ceux qui possèdent déjà un Syquest.

La fameuse carte "Live" et son logiciel "Invision plus", permettant des digitalisations ou des manipulations vidéo en temps réel, sera enfin disponible à la fin février de cette année. Ceux qui s'impatientaient se réjouiront d'une baisse non négligeable de prix, offrant l'ensemble à moins de 8000 F. De plus, ils profiteront d'une version française qui fonctionne directement sur A3000.

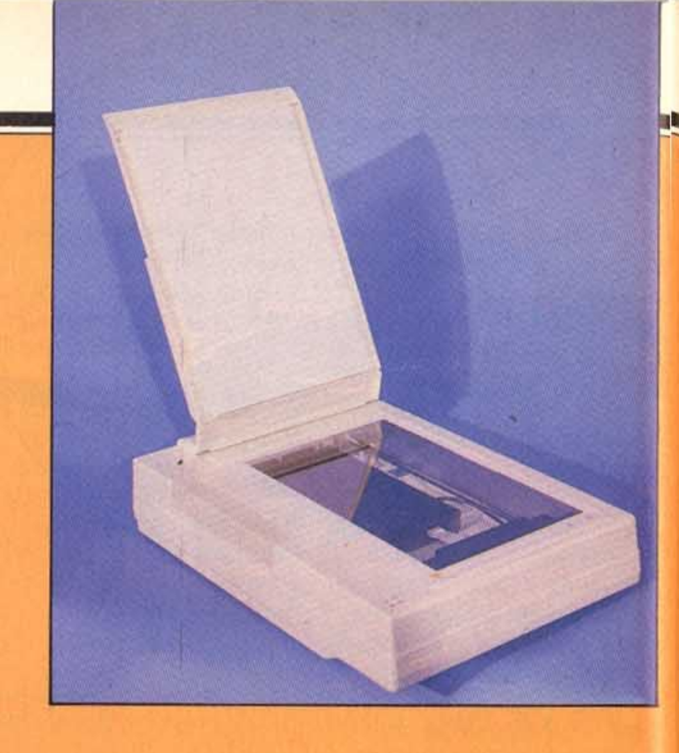
La dernière des Potaings Bordelais nous informe que CIS, importateur de matériel informatique, a déménagé à l'Europarc, situé 14 avenue Gustave Hertz, 33 600 Pessac.

ALLO ?

Les produits Print Technik sont depuis peu commercialisés en France, via Bus+ qui propose pour commencer deux modèles de scanners.
Le premier est tout crème, vous aviez déjà pu le découvrir dans les pages de ST Magazine. Il offre une résolu-

tion de 600 dpi, avec 64 niveaux de gris. Il peut fonctionner sur tout type d'Amiga, du A500 au A3000. Il est fourni avec OCR (reconnaissance de caractères), pour environ 9500 F.

Le second est tout gris et possède une grande nouveauté. Il remplit aussi le rôle de Fax, en plus des fonctions de scan, de photocopie et d'impression thermique. Un splendide téléphone rouge en plastique est disponible en option, muni d'un connecteur spécial standard téléphonique USA. Un adaptateur vous permettra d'utiliser votre propre téléphone.
Le Fax a une vitesse de transfert de 9600 bauds G3 tout à fait standard. Il nécessite toutefois l'installation d'une deuxième ligne téléphonique pour profiter au mieux du Fax, sans perturber les communications courantes.





Le logiciel du scanner fourni est simple vis-à-vis des résultats. Une option permet de transférer directement une image scannée à DeluxePaint III pour les éventuelles retouches. La qualité du résultat est bonne en tenant compte d'une résolution de 200 dpi (points par pouce) avec 16 niveaux de gris.
On dispose également d'un émulateur Nec P6, permettant l'émulation d'une imprimante graphique, pour obtenir en 180 dpi une page A4 en 9 secondes. Il faudra tenir compte du prix relativement élevé du papier thermique. La petite bête est annoncée aux alentours de 9900 F.

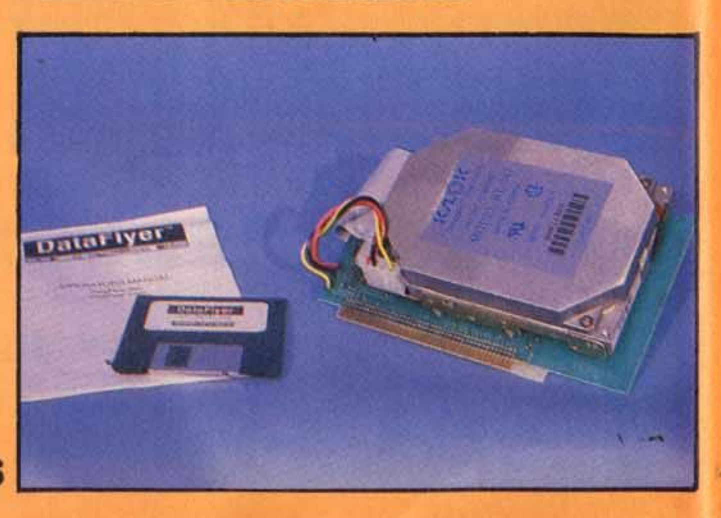
UN DE MOINS...

Comme toute machine récente, l'A3000 souffre encore de petits machins qui font que... Par exemple, les derniers moniteurs multisync 1950-B fournis par Commodore avec les A3000, sont réfractaires au genlock. C'est gênant, mais pas irrémédiable. SATV a en effet conçu un adaptateur spécifique pour résoudre ce petit problème. Fini d'avoir les yeux dans les coins, il suffit de l'intercaler entre la sortie vidéo de votre A3000 et le cordon du moniteur.

LA BIBLE

La Bible de l'Amiga, des Éditions Micro Application, vient de subir un lifting. Pour ceux qui ne connaissent pas encore cet ouvrage de référence, sachez que vous pouvez y trouver, parmi ses 1110 pages (!), la description complète de tous les Amiga, dont celle de l'Amiga 3000, qui justifie à elle seule cette importante mise à jour. Les circuits spéciaux, tels que Agnus ou Paula, ainsi que le hardware général de la machine sont décrits de façon explicite, grâce à de nombreux schémas. Le software n'est pas en reste, car heureusement, et c'est là son but, la Bible de l'Amiga comporte aussi la description détaillée de toutes les fonctions des différentes librairies du système, ainsi que leurs paramètres, que ce soit pour ouvrir une fenêtre sous intuition ou pour s'amuser avec un DisplayAlert. Il suffit de rechercher le vecteur dans la librairie concernée pour trouver toutes les informations qui lui sont relatives. De nombreux exemples, en langage C et en Assembleur, parsèment cet ouvrage et permettront de vous guider dans vos efforts créatifs, sans qu'un guru rageur ne vienne vous tirer les oreilles. Pour les courageux, et néanmoins inconscients, programmeurs qui désirent quitter le relatif confort du système et piloter les Custom Chip's à haute vitesse, une description détaillée des différents registres bit par bit leur est proposée. Pour conclure, sachez que cet ouvrage exceptionnel est une source de renseignements intarissable.
Prix 340 F, Éditions Micro Application

PREMIER PRIX MALIN...



BUS+, comme à l'habitude, sait trouver le petit plus, et cela devient très ardu lorsqu'il s'agit d'une carte contrôleur SCSI et de son disque dur : un prix très attractif, c'est la moindre des choses, et un produit de bonne qualité, élémentaire pour l'importateur d'IVS...
Le petit truc malin de la DataFlyer 2000 est d'être installée à l'envers de toutes ses concurrentes. L'unité de disque dur se trouvant sur l'autre face permet d'installer la carte sur le slot le plus à gauche d'un A2000. Ça sert à quoi ? Elle évite de bloquer deux slots inutilement à cause de la simple épaisseur du disque dur. Son prix est inférieur à 4000 F.

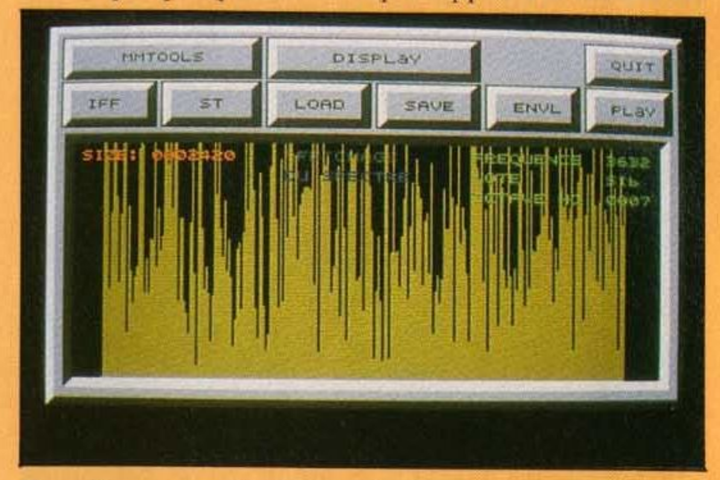
3D PROFESSIONAL : "LE" Ray Tracer

Dans le précédent Cahier Amiga, lors du test de la première version de 3D Professional, nous vous informions que l'équipe de Cryogenic Software procédait à l'optimisation d'un module de RAY TRACING externe à 3D Professional. La version 2.0 du soft vient de pointer son nez, et elle inclut le suscité module, qui brillait par son absence lors du premier test. Quoi de neuf Docteur ? Eh bien, quelques améliorations notables du côté du modeleur, avec notamment une option "World". Le Ray Tracer quant à lui, sujet de cette news, semble répondre à notre attente, fidèle à son modeleur. Le nombre de paramètres proposé est édifiant : profondeur d'analyse des rayons lancés, prise en compte des ombres portées, de l'éclairage spéculaire, du nombre de réflexions maximum, etc. La possibilité de modifier l'atmosphère ambiante permet des effets spéciaux intéressants tels que la présence de brouillard ou de poussière. Des fichiers "Batch" permettent de calculer plusieurs images à la suite de façon automatique. Toutefois, il faut que vous sachiez que, ce module de rendu étant très pointu, le temps de calcul nécessaire est réellement préjudiciable à une utilisation intensive sans carte accélératrice.



MUSIC MASTER

Février verra la sortie de Music Master, un logiciel pour A500 et A2000 dédié à la création musicale, réalisé par Computer's Dream, et importé par Ubi Soft. La version ST étant, si je ne m'abuse, déjà traitée dans nos pages, nous n'allons pas tomber dans le rabâchage dû au gagatisme précoce. Petit résumé, Music Master permet de créer ou de retravailler des échantillons sonores, d'enregistrer des mélodies grâce au clavier de l'Amiga ou à un instrument MIDI, et d'organiser le tout pour former un morceau complet. Music Master Amiga possède néanmoins quelques particularités par rapport à la version ST.



Il ne dispose que de 4 voies (au lieu de 16 sur ST), car il utilise les possibilités sonores internes de l'Amiga et non une cartouche externe. On peut cependant bénéficier d'une cinquième voie comme buffer, lors de la composition par exemple, ou pour copier des blocs.
On trouve aussi 4 voies destinées à créer des effets sonores. On réalise ainsi des "shuntes" en modulant le volume de la voie. On peut également faire varier la période du son à chaque temps. Autre type d'effet, il est possible de reboucler un son sur lui-même.
MMTools, le programme de conversion, permet de récupérer des sons au format IFF (Sound Tracker), et ST-Replay. De plus, il calcule automatiquement la "transformée de Fourier" du son importé, et affiche son spectre, avec indication de la note, de l'octave, de la longueur, et de la fréquence fondamentale. On peut également y créer une enveloppe de volume, que l'on pourra récupérer par la suite à l'intérieur du programme principal.

Utilitaire de conversion des sons IFF



Lors de l'enregistrement, on bénéficie du témoin audio des trois autres pistes. Le clic sonore est remplacé par un flash de l'écran.
Quoi d'autre encore, le logiciel est protégé par un dungle (bitonniau) à insérer sur le port joystick. Pour le reste, vous pouvez toujours aller admirer la couverture (la photo est issue du package Music Master) en attendant le banc d'essai complet dans un prochain numéro.

NORDIC POWER



Vous aviez déjà fait connaissance avec la Nordic Power pour A500 dans un numéro de ST Mag. Comme chacun sait, la cartouche tourne en tâche arrière sur votre machine, et regarde tourner les programmes comme une vache regarde passer les trains. A la grande joie de tous, une version est désormais disponible pour A2000, merci Bus+. Petite précision d'installation, la Nordic Power est munie d'un connecteur 86 broches, qui doit être inséré dans l'un des connecteurs 100 broches de l'Amiga, et non dans le connecteur 86 broches MMU de la carte mère (attention à la "gourance"...).
Après avoir arrêté un programme en cours d'exécution, vous disposerez de nombreux outils pour sauvegarder une portion ou l'ensemble de la mémoire, créer des répertoires, effacer, formater, lui subtiliser ses secrets, ses images, ou ses sons, et vous pourrez même le relancer après toutes ces indiscrétions...
La cartouche peut aussi tourner avec un Super Fat Agnus. Elle supporte même la présence d'une carte passerelle PC dans votre Amiga, mais elle ne peut fonctionner qu'en mode Amiga. La Nordic Power vous rendra bien des services au moment du débogage, ou pour toute autre activité nécessitant "l'espionnage" du système et de la mémoire.

BANDE ANNONCE...

Depuis le temps que l'on en parle... Le voilà, on le présentait officiellement à Las Vegas le 9 janvier 1991. En France, on nous parlera bientôt du CDTV de chez Commodore. En version américaine, ce fabuleux objet vous permet de goûter chez vous les joies du multimédia interactif.

Sur le plan technique, en attendant la publication le mois prochain dans ST Mag des spécifs complètes, il faut savoir que l'on constate une énorme similitude avec les spécifications d'un Amiga 500 doté d'un lecteur de CD-ROM, capable de lire aussi bien un compact disque audio que logiciel. Mais que peut-on faire avec un Amiga 500 sans clavier ? Pas grand-chose...
On trouve quelques accessoires en option, tels que :
- lecteur de disquettes ;
- trackball infrarouge ;
- joystik ;
- carte d'interface MIDI ;
- cartes personnelles RAM ou ROM ;
- genlock ;
- interface infrarouge pour deux joueurs ;
- modem ;
- imprimante ;
- et, vous ne me croirez pas, un clavier, youpi !...

Mais voyons voir avec la télécommande à infrarouge, ce qu'il nous reste à faire...
On pourra jouer : il vous faudra choisir dans la trentaine de titres disponibles dès le lancement, voire une centaine d'ici la fin de l'année, parmi les marques les plus connues, telles que LucasFilm, Psygnosis, Guiness, Cinemaware, Accolade, Virgin...
On pourra s'instruire : le disque simple de l'Encyclopédie Grolier remplace les 21 volumes de son homologue imprimée, en fait 30000 articles illustrés de 2000 images. Vous trouverez aussi le World Vista Atlas ou alors l'American Heritage Encyclopedic Dictionary.
On pourra apprendre : les petits découvriront la lecture ou la compréhension des nombres grâce aux productions Disney Software, en retrouvant Cendrillon, Peter Rabbit...

Voilà, ceci est très prometteur, aux Etat-Unis surtout. Le CDTV sera-t-il en France accessible à tous les publics, les titres seront-ils tous francisés, quelle sera la production française, quelle application en feront les programmeurs les plus inventifs ? Vous le saurez prochainement sur ST Mag...



PAINTER 3D

À LA RECHERCHE DU TEMPS RÉEL

Présenté sur le stand GVP lors de l'AMIGA 90 de Cologne, Painter 3D d'Adept Development est un nouveau logiciel de création en trois dimensions. Il correspond à la tendance qui anime les développeurs en ce moment, à savoir la convivialité, l'ergonomie et le temps réel. Ses concepteurs, suisses, sont l'illustration d'une deuxième tendance selon laquelle la recherche en matière de logiciels de 3D n'est plus seulement l'apanage de nos voisins du Nouveau Monde.

ASPECT

Le package de Painter 3D se compose d'un module de modélisation, d'animation et d'une disquette d'objets et d'images. N'ayant pu obtenir que l'outil de modélisation, je me cantonnerai à vous donner un descriptif et mes impressions sur ce que l'on a bien voulu nous fournir.
La documentation fort bien réalisée est en français, ce qui est appréciable lorsqu'on est en présence d'un logiciel étoffé, et puis ça nous change de l'habitude. Dès le lancement du programme, on se demande immédiatement si l'on est bien dans un soft de 3D tant l'interface utilisateur ressemble à celles que l'on rencontre en 2D. On dispose sur la droite de l'écran de plusieurs blocs d'icônes à la DeluxePaint. Dans la barre de menu, seuls les indispensables charge, sauve, quitte, etc., sont présents.

LES CONTOURS

Avant de générer un objet en volume, Painter 3D nous invite à dessiner sa base ou contour, dans un plan. Dès la validation de cette option, quatre blocs d'icônes supplémentaires apparaissent dans l'espace de dialogue et une fenêtre dans l'écran principal jusqu'ici vide. Une option de grille aimantée paramétrable aide à tracer un contour précis et surtout de vraies lignes droites. On crée un segment de droite, puis en cliquant de nouveau le bouton de la souris, un triangle, et ainsi de suite jusqu'à un polygone très irrégulier mais toujours dans un périmètre fermé. Un segment de deux points peut se transformer en arc de cercle ou en spline ajustable en déplaçant la souris. C'est très pratique, et plus convivial que dans Sculpt, car on n'a pas à se préoccuper du nombre de points de la courbe (petite précision pour les connaisseurs qui apprécieront). Si l'exécution ne vous plaît pas, la fonction d'annulation "undo" omniprésente annulera la dernière action pour revenir à la précédente (il



Le Modeleur : en mode "solide", on distingue les sous-objets qui forment letapis. De même, le salon n'est constitué que d'un seul fauteuil dupliqué.

y a même des logiciels qui n'en ont pas, suivez mon regard...). On dispose de tous les outils nécessaires au dessin, axes, zoom, polygones réguliers (donc cercles), rotation, déplacement de points, miroirs, rotation du contour, etc. Il est permis de copier un contour et de travailler en même temps sur plusieurs fenêtres. Des paramètres en permanence affichables renseignent sur le nombre de points, l'échelle, la grille, la position des axes, etc. De même, des coordonnées polaires et cartésiennes vous aiderons à affûter la précision du dessin.

LE VOLUME

On peut générer un volume de plusieurs façons dans Painter 3D. D'après un contour dessiné au préalable, on procède à une extrusion linéaire pour créer un prisme droit ou bien pas d'extrusion du tout, le contour sera considéré comme une simple facette plane. Il est possible de fabriquer des pyramides ou des troncs de pyramide et des objets de révolution grâce à une fonction "spin". La mise en volume réalisée, on peut faire tourner l'objet sur lui-même, le voir sous tous les angles en mode fil de fer ou en faces cachées.

LES VUES

Nous sommes maintenant au coeur du logiciel ; en cliquant sur l'icône "vue", une nouvelle fenêtre apparaît dans laquelle nous allons insérer les objets créés. Deux modes de projection sont possibles, en perspective et en projection orthogonale, cette dernière correspond plus à une approche d'architecte face à un plan. Une fenêtre active peut être copiée avec les mêmes caractéristiques de paramètres, icônes actives, etc. C'est pratique quand on veut faire des modifications et conserver un original afin de comparer les changements. Les commandes sont celles que l'on retrouve dans tous les logiciels 3D : zoom, déplacements, rotations des objets et de l'observateur, selon des axes que l'on prédéfinit dans le plan ou en perspective.

Signalons quelques originalités : on peut couper une scène pour voir ce qui se passe derrière un groupe d'objets, ou dire à l'observateur de se déplacer ou de s'aligner par rapport à un point indiqué par un clic souris. Toutes les actions sont accessibles par les icônes et paramétrables en distances (millimètre, mètre, kilomètre), angles de rotation, facteurs d'échelle et de focale.

LES SOUS-OBJETS

C'est avec les sous-objets que Painter 3D diffère des autres logiciels. En effet, l'idée est d'utiliser des objets existants pour en composer un plus complexe, qui deviendra à son tour une partie d'un autre objet, et ainsi de suite. La mémoire utilisée par l'ordinateur n'est en fait occupée que par les éléments de base. On peut ainsi créer un assemblage d'une infinité d'objets avec leurs propres attributs de taille et de couleur. C'est assez impressionnant surtout dans les animations, par exemple quand vous voulez qu'une sphère explose en une multitude de petits cubes.

couleur. On a le choix parmi douze couleurs simultanément, c'est peu, mais cela est dû au type de rendu d'image proposé par Painter 3D, nous verrons cela plus loin. La fenêtre de la palette est quasiment identique à celle de DeluxePaint avec les mêmes options, y compris le cyclage des couleurs.

Une scène est éclairée par trois types de sources lumineuses : la lumière ambiante, les sources directives (des rayons provenant de l'infini) et les spots (ronds de lumière). Ces trois sources sont modifiables en intensité et en saturation, mais pas en couleur qui reste blanche. Une particularité cependant, les spots sont liés aux objets ; un objet peut comporter un nombre quelconque de spots, qui seront dupliqués si l'on crée des sous-objets à partir de celui-ci. Dans une scène comportant une table et un vase dans une pièce, seul le vase peut être éclairé si la pièce et la table n'ont pas de spots qui leur sont rattachés.



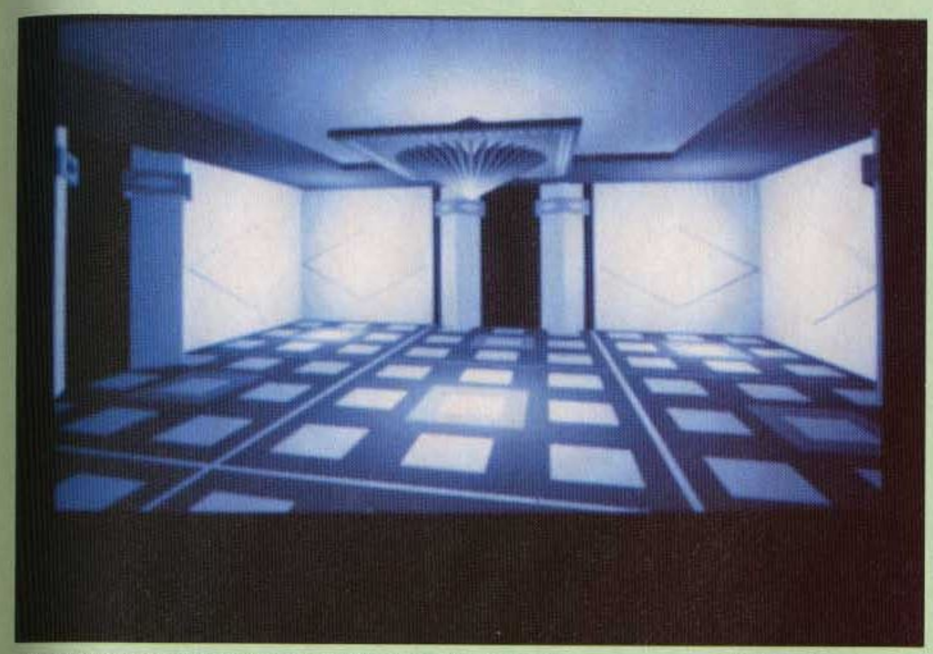
Rendu 16 couleurs Interlace : chaque pavé du sol a un spot qui lui est rattaché. Malgré les 16 couleurs, l'effet d'ombrage de la trame apporte une texture intéressante.

LA COULEUR ET L'ÉCLAIRAGE

Pour modifier la couleur d'un objet ou d'un sous-objet, il suffit de cliquer dessus pour le sélectionner, et d'appeler la palette pour changer la

LE RENDU

L'image calculée par Painter 3D est réalisée sous la forme d'un tramage pour obtenir des dégradés. Il n'y a pas de mode HAM (4096 couleurs)



Rendu en mode Filtre : exclusivement monochrome, on appréciera la qualité de l'antialiasing. Cette image est récupérable dans DeluxePaint.

propre à l'Amiga, les images sont donc récupérables sous DeluxePaint dans tous les types de résolution. La fenêtre active dans le modeleur devient le cadrage final avant le lancement du calcul. Il existe un mode de calcul "filtré" où l'effet "marches d'escalier" est atténué, pour ne pas dire supprimé, mais l'image sera monochrome, c'est pourtant assez réaliste. Je disposais d'un logiciel standard, non compilé pour un 68030, et le temps de calcul m'a paru relativement long, reste à tester si cela est toujours vrai avec une version pour machine gonflée.

SYNTHESE

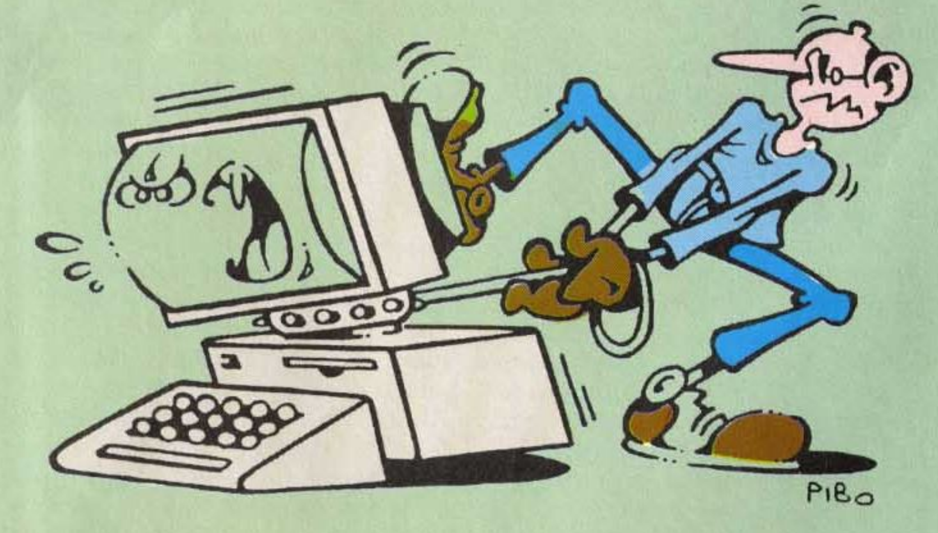
Painter 3D est-il un bon logiciel ? Voilà toute la question, je vous donnerai donc mon avis subjectif, sachant que ce qui est bon pour l'un ne l'est pas forcément pour l'autre. Ses qualités sont dans la rapidité et la précision avec laquelle on modélise. Le principe des sous-objets est très puissant quand on arrive à le maîtriser, et c'est là que se situe le gros intérêt et l'originalité de Painter 3D.

L'interface utilisateur avec ses icônes est intéressante, mais plus "fouillis" que celle de Caligari, qui reste quand même la référence dans ce domaine. On peut également déplorer l'absence de textures, de mapping d'images 2D, ainsi que le mode de rendu qui ne laisse pas le choix.

Il reste à voir si le module d'animation apporte un plus par rapport aux autres logiciels. La solution à ces carences est détournée par les programmeurs, qui proposent un module de transfert des objets vers Sculpt-4D et vice versa (via Interchange, on pourra accéder à Turbo Silver pour le mapping). Par contre, on ne peut pas tirer parti de la multitude des sous-objets, seuls les objets élémentaires seront transférés dans les autres softs, dommage...
Suis-je sévère dans mon jugement ? OUI, quand on pense que c'est le premier jet de ce logiciel et que la suite sera sans doute prometteuse ; NON, si ce logiciel a des prétentions de logiciel professionnel.

Frédéric Potier





À L'AISE...

Il y a bien des façons de communiquer avec votre Amiga préféré : conversation en tête-à-tête dans l'intimité d'un shell, taquinage d'icônes à la souris, branlage de sprites au joystick, ou coups de pied intempestifs, le tout est d'adapter le geste à la situation.
Easyl est une tablette graphique, encore une autre forme de capteur peut-être plus proche du geste naturel de la main. Avec n'importe quel crayon ou bout de machin pointu, vous pourrez remplacer la souris pour écrire, dessiner, à l'intérieur de la plupart des logiciels Amiga. Pas tout à fait nouveau, ce produit d'Anakin Research n'était pas jusqu'à présent parvenu jusqu'à nous. Sur l'initiative de Bus+, la tablette est maintenant disponible en France, avec sa notice francisée

Le package complet regroupe la tablette, son interface hard, une disquette et la notice. En premier lieu, l'interface se place dans un slot du 2000. Le câble de la tablette, muni d'une DB9 à son extrémité, doit être ensuite relié à l'interface sur la prise correspondante. La disquette contient quant à elle les drivers de la tablette, pour l'utiliser sous d'autres logiciels, et un programme de dessin.

LA TABLETTE

La tablette est en plastique résistant, d'une couleur crème assortie à l'Amiga. La surface intérieure (21,5cm x 33cm) est entourée d'un large rebord (surface totale 34cm x 45cm), rendant son utilisation très confortable.
Sa résolution est de 1024x1024 points. Cependant, la surface utile est réduite à 1024x944, en raison de la place nécessaire à la bande de sélection. Sa vitesse d'échantillonnage est de 250 paires de coordonnées (x et y) par seconde. Les deux boutons sur le côté de la tablette représentent les boutons de la souris.
Pour dessiner, une feuille de papier A4 peut être fixée par un adhésif double face. Si l'on désire repositionner avec précision une feuille, il est possible d'installer des chevilles (fournies), et d'employer du papier perforé. Easyl peut également recevoir des barres de repère couramment utilisées dans les techniques d'animations traditionnelles, comme la superposition de calques.

LES DRIVERS

Le driver est indispensable pour utiliser la tablette sous d'autres logiciels Amiga. Il suffit de lancer un driver, puis le programme de votre choix. Un seul driver à la fois peut être présent dans la mémoire, et il est désactivable à tout moment par combinaison de touches.
Plusieurs types de driver sont proposés. Le plus simple ne nécessite que peu de mémoire. Avec le driver "pro", vous pouvez dessiner une image de la dimension totale de la tablette dans une portion de l'écran Amiga. Une fenêtre, dimensionnable numériquement ou à la souris, permet d'en définir avec précision la taille et la position. Les proportions du dessin sur la tablette seront respectées à l'écran, tant que les rapports tablette/écran de hauteur et de largeur restent à peu près équivalents. Plus les rapports sont différents, plus la déformation est importante. Pour le normal comme pour le pro, un autre driver spécifique est disponible, qui simule l'enfoncement continu des boutons de la souris. Cela s'avère utile dans certains modes de dessin. Les gauchers n'ont pas été oubliés, puisque les mêmes drivers leurs sont proposés, mais la tablette retournée (boutons souris à droite pour dessiner de la main gauche).

Si vous utilisez fréquemment le même logiciel avec la tablette, vous pouvez créer ce que l'on appelle un "gabarit", afin d'optimiser le confort de travail. Il vous suffit de repérer l'emplacement de toutes les fenêtres, icônes, et diverses commandes employées par le logiciel, et de les reproduire sur une feuille de papier. Placé sur la tablette, ce schéma vous indiquera l'endroit exact à valider, et vous évite ainsi de tâtonner ou de lâcher le crayon pour reprendre la souris.

LE LOGICIEL

La tablette possède son propre logiciel de dessin. Pour le lancer, pas besoin de driver, il suffit de cliquer sur l'icône du programme Easyl. Il est prévu pour fonctionner de pair avec la souris, pour l'accès aux menus déroulants, la manipulation des fichiers images, ou le choix des pinceaux par exemple.

Le logiciel peut importer et sauvegarder indépendamment une image et une palette. Il supporte quatre résolutions différentes, de 320x256 à 640x512, qui n'influent en aucun cas sur la résolution de la tablette elle-même. Pour plus de confort, les pointeurs (crayon et souris), ainsi que la barre de menus, sont escamotables.



L'orientation de la tablette est variable. Les gauchers pourront retourner la tablette, comme nous l'avons vu plus haut. Les portraitistes la préféreront verticale, avec en option la visualisation de la surface utile sur l'écran.

Des outils de dessin, ainsi que le choix des couleurs, sont disponibles directement sur la tablette, en appuyant le crayon dans le carré correspondant. Après avoir fait un beau gribouillis, l'on dispose d'une fonction de remplissage. Si c'est un chef-d'oeuvre, "Keep" conserve provisoirement le dessin dans une page arrière, et "Restore" le restitue à volonté. Si c'est une horreur, "Clear" le renverra dans le monde de l'oubli.
Dans le menu, on détermine la forme du pinceau (pixel, ligne, carré, pointillés...). On peut également avoir accès à la palette pour modifier les couleurs de base. Par défaut, on dispose de 32 couleurs en 320x200, et de 16 couleurs en 640x200.

Easyl offre un tracé très souple. En mode trait, l'affichage à l'écran suit correctement le mouvement de la main, ce qui permet de reproduire fidèlement une signature par exemple. La distorsion de l'image est réduite, l'homothétie des formes est respectée (tant qu'aucune déformation n'est volontairement paramétrée).
Cette tablette devrait convenir à la saisie au vol de croquis, schémas, "roughes" en tous genres, pour capturer en temps réel une idée dans l'Amiga. Ces images pourront ensuite être réimportées dans d'autres logiciels de dessin pour retouche et finition. Easyl peut également satisfaire d'autres types de graphismes, notamment la modélisation d'objet, le dessin animé,... Beaucoup d'autres applications sont bien sûr réalisables, il est même possible d'avoir les sources pour développement. Les novices, eux, apprécieront sa simplicité, et profiteront du plaisir de dessiner.
La notice semble un peu désuette, mais, me dit-on en régie, une mise à jour est prévue pour très bientôt. Enfin, un p'tit mot du prix, ça devrait le faire à moins de 3000 F.

Anne Olivelli

LE FORMAT IFF (II)

Comme promis le mois dernier, nous nous retrouvons de nouveau pour une nouvelle découverte de l'IFF. Ce mois-ci, nous quittons le domaine du graphisme pour nous diriger à grands pas vers la musique, et plus précisément vers le format 8SVX, le seul et l'unique.

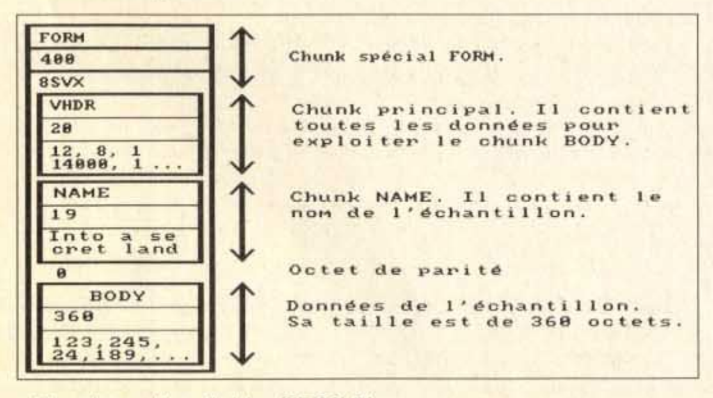
Sous ce nom quelque peu barbare se cache en fait un format permettant la gestion d'échantillons sonores et de fichiers d'instruments de musique. En effet, le terme 8SVX signifie littéralement "8 Bits Sampled Voice". Mais avant de passer à son étude détaillée, je vais procéder à un rappel des notions de base à connaître pour ceux qui prendraient le bateau en cours. Pour plus de renseignements, se reporter au numéro précédent. Dans toutes les définitions de structure, j'utiliserai la syntaxe suivante :

```
UBYTE = 1    octet non signé     8 bits
WORD =  1    mot signé          16 bits
UWORD = 1    mot non signé      16 bits
LONG  = 1    mot long signé     32 bits
```

Un chunk est un bloc de données. Cela signifie que sa structure d'identification (ne pas confondre avec sa structure interne de données) sera toujours la même et pourrait être définie ainsi en C :

```
typedef struct {
  ID     ckID;        /* 4 caractères d'identification */
  LONG   ckSize;      /* Taille du chunk en octets     */
  UBYTE  ckData;      /* Données                       */
  } chunks;
```

Attention : les quatre caractères d'identification doivent impérativement être en majuscules et le premier octet des données doit impérativement se trouver à une adresse paire relative par rapport au début du fichier. Pour cela, on peut utiliser, si besoin est, un octet nul (Pad-byte), aussi appelé octet de parité. Ces petits rappels effectués, nous pouvons maintenant passer aux choses sérieuses.



Structure d'un fichier IFF 8SVX

Pour générer un fichier IFF 8SVX, il faut bien sûr disposer d'une interface (celle qui fait hurler grand-mère est très bien) permettant la liaison entre l'Amiga et la source sonore (platine laser, platine cassette, etc.) et d'un programme capable de retraiter l'échantillon et de le sauvegarder au bon format (Audiomaster II, par exemple). Comme pour le format IFF ILBM, il existe bien entendu différents chunks.

- VHDR (Voice header) :
Ce chunk, d'une taille de 20 octets, contient toutes les informations permettant de gérer le chunk BODY. Voici sa structure en C :

```
#define sCmpNone        0 /* Pas de compression             */
#define sCmpFibDelta    1 /* Compression Fibonacci-delta */

typedef struct {
  ULONG oneShotHiSamples; /* Taille de l'échantillon =  */
  ULONG repeatHiSamples;  /*     ( oneShotHiSamples +   */
                          /*          repeatHiSamples  )*/
  ULONG samplesPerHiCycles; /*Nbre d'échantillons/cycle */
                          /* dans la plus haute octave.*/
  UWORD samplesPerSec;     /* Fréquence d'échantillonnage /
  UBYTE ctOctave;         /* Nbre d'octaves dans le fichier */
  UBYTE sCompression;     /* Compression utilisée ?        */
  LONG volume;            /* Volume sonore de référence. Il */
           /*  est utile si ATAK et RLSE sont définis.*/
  } Voice8Header;
```

Les champs "oneShotHiSamples" et "repeatHiSamples" permettent de connaître la taille de l'échantillon dans la plus haute octave (taille = oneShotHiSamples + repeatHiSamples). Ces deux champs n'ont bien évidemment pas que cette fonction. Ainsi, pour restituer un échantillon, vous devez utiliser le contenu de "oneShotHiSamples", puis répéter un certain nombre de fois la partie contenue dans "repeatHiSamples". Cette dernière pouvant être alors comparée à une sorte d'écho.

Le champ "samplesPerHiCycles" indique le nombre d'échantillons par cycle dans la plus haute octave (dans le cas d'un fichier instrument) ou bien la valeur 0 si cela est inconnu. Chaque octave successive (donc une fréquence inférieure) contiendra deux fois plus de données. Par exemple, si la taille des données de la plus haute octave est de 10000, la taille de l'octave précédente sera de 20000.

Le champ "samplesPerSec" définit quant à lui la fréquence d'échantillonnage. Attention, cette valeur est très importante lors du traitement du son. En effet, si vous la gérez mal, vous risquez d'obtenir un effet quelque peu particulier. Essayez donc d'échantillonner un bon chanteur (ça existe encore) à 10000 Hz et écoutez-le à 14000, vous aurez presque l'impression d'entendre la voix de Vanessa P. (comme quoi nous sommes bien peu de choses...).

Le champ "ctOctave" indique le nombre d'octaves contenues dans le chunk BODY. Cette valeur est seulement utilisée dans un fichier instrument, le seul à pouvoir contenir plusieurs octaves.

Le champ "sCompression" indique si le chunk BODY a subi une compression de données ou pas. Actuellement, la seule compression utilisée est celle de Fibonacci. Elle permet en effet de diminuer de moitié la taille d'un échantillon, mais elle entraîne en retour un effet de distorsion. Attention, si vous voulez créer votre propre player, pensez qu'il vous faudra gérer vous-même le décodage du chunk BODY avant de pouvoir l'utiliser. Cela implique donc une réservation de mémoire d'une taille deux fois plus grande que celle de l'échantillon.

Enfin, le champ "volume" contient le volume de référence. Il faut savoir que cette valeur sera celle qui servira de base au cas où les chunks ATAK et RLSE seraient utilisés. Je vous donnerai plus de détails lors de leur étude respective.

Je vais maintenant donner une rapide description des 4 chunks qui vont suivre. En effet, ceux-ci ne contiennent que du texte (dans la limite de 32767 caractères par chunk) et ne servent en rien au décodage du chunk BODY.

- NAME

Ce chunk peut contenir le nom de l'échantillon quand il est indiqué (par exemple, "Into a secret land").

- (C)

Ce chunk peut contenir un copyright sur l'échantillon (par exemple, "© TVG industries 1991").

- AUTH

Ce chunk peut contenir le nom de l'auteur/créateur.

- ANNO

Peut contenir une annotation quelconque de l'auteur.

- ATAK (Attaque)

Permet de définir l'attaque du son. Voici sa structure :

```
typedef struct {
  UWORD duration;  /* Durée en millisecondes           */
  LONG dest;       /* Facteur de calcul pour le volume */
  } EGPoint;
```

Il faut savoir qu'un codage particulier est utilisé. En effet, ce sont des points (EGPoints) qui sont employés. Chaque point représentant un nouveau changement de volume d'une durée de "duration" et le dernier EGPoint définissant la valeur du Sustain.

Le champ "duration" indique la durée entre 2 points en millisecondes, tandis que le champ "dest" indique, quant à lui, le facteur de multiplication à affecter au volume indiqué dans le chunk VHDR. Le calcul du volume se fait de la façon suivante : Volume final = 'volume' * 'dest'.

- RLSE (Release)

Ce chunk permet de définir le déclin du son. Sa structure est identique à celle du chunk ATAK, et le même codage est utilisé (voir plus haut). Chaque EGPoint représentant un nouveau changement de volume, le dernier EGPoint définissant le volume final.

Le champ "duration" indique la durée entre deux points en millisecondes, tandis que le champ "dest" indique le facteur de multiplication à affecter au volume indiqué dans le chunk VHDR.

Les deux chunks ATAK et RLSE contiennent donc à eux deux l'enveloppe du son, soit l'attaque, le sustain et le déclin (release).

- BODY

Ce chunk, comme pour le format IFF ILBM, contient l'échantillon sonore. Attention, ce chunk possède lui aussi une structure spéciale. En effet, il est divisé d'une manière bien particulière : on trouve d'abord les octaves (celles de plus haute fréquence venant en premier, les autres après), puis chaque octave est elle-même divisée en 2 parties (la 1ère étant la "oneShot", la 2ème la "repeat" - voir structure VHDR), enfin viennent les cycles.

Enfin, pour tous ceux qui viennent, ou qui vont investir dans une boîte du style de celle qui fait hurler grand-mère (et même pour les autres), quelques conseils pour obtenir des échantillons à rendre jaloux votre voisin qui vient de s'offrir la dernière platine SYNO modèle 12XF180 64 Bits converter 24 times oversampling digital filter...

1) Au risque de passer pour un idiot (je prends vraiment tous les risques), je précise que la source musicale doit être la meilleure possible au départ si l'on veut obtenir quelque chose de correct à l'arrivée.

2) Pour obtenir un bon échantillon, il faut l'échantillonner à deux fois sa plus haute fréquence. Je m'explique. Si la plus haute fréquence de ce que vous voulez échantillonner est de 5000 Hz, vous devrez utiliser une fréquence d'échantillonnage de 10000 (5000 x 2) ;

3) Il faut aussi savoir que le filtre passe-bas de l'Amiga, quand il est activé, supprime toutes les fréquences supérieures à 7000 Hz. Il est donc inutile d'utiliser une fréquence supérieure à 14000 Hz ;

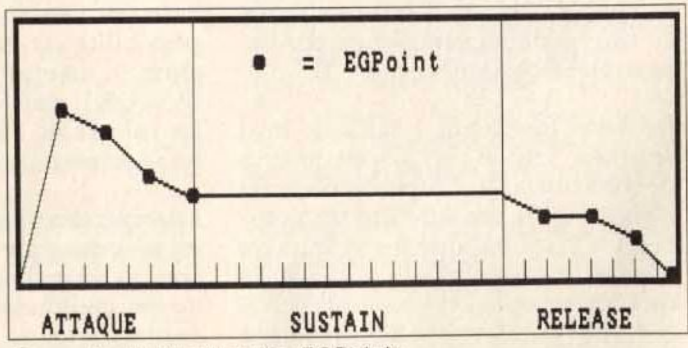
4) Si vous voulez définir des points de répétition dans votre échantillon, méfiez-vous des valeurs trop différentes les unes des autres qui engendrent un léger "clic" toujours désagréable à l'écoute ;

5) Sachez qu'il est souvent inutile d'échantillonner en stéréo. En effet, cela n'apporte que bien peu de choses, mais par contre, la taille de l'échantillon est doublée ;

6) Enfin, essayez d'avoir le maximum de puissance pour éviter le souffle, mais samplez à un niveau assez bas pour ne pas engendrer de distorsions (si votre sampler le permet).

Et voilà, maintenant que vous savez tout sur le format IFF 8SVX, je vous donne rendez-vous au mois prochain pour l'étude du format ANIM, si tout se passe bien. Toutefois, si vous avez des remarques ou des questions à formuler, n'hésitez pas à m'écrire, je verrai ce que je peux faire pour vous.

T.G.V.



Exemple d'utilisation des EGPoints

ANIMATIONS GRAPHIQUES EN ASSEMBLEUR (III) : LE BLITTER

Après avoir vu les processeurs "passifs" de l'Amiga, nous passons aujourd'hui au seul dans la machine, 68000 mis à part, qui a la capacité de faire des transferts de mémoires. Il s'occupe de bouger les fenêtres d'intuition, les bobs des jeux d'arcade, et même (sacrilège !) de décoder les données du drive : il s'agit du Blitter.

Jusqu'ici, nous nous sommes restreint à un type de programmation graphique qui demande très peu de temps machine, car nous utilisions les possibilités de l'affichage luimême ; en effet, on peut faire des scrollings en jouant avec l'adresse de la mémoire vidéo et le delay, et l'on peut faire un déplacement de dégradé de couleurs en ne décalant que deux octets par ligne, soit en général moins d'1 ko de données. On ne modifiait pas la mémoire vidéo elle-même. Mais si l'on a besoin d'afficher rapidement une image, n'importe où sur l'écran, pardessus une autre, en la positionnant au pixel près, comment faire ? Il faudrait utiliser le 68000 et transférer une grosse quantité de données en décalant binairement chaque octet, opération pour laquelle le 68000, malgré tout le respect que je lui dois, n'est absolument pas qualifié. L'Amiga est donc muni d'une puce spécialisée, le Blitter, qui exécute à la perfection ce genre de tâche ingrate. L'utilisation du blitter est simple, comme le Copper, il se programme au travers de registres situés dans la zone $DFFxxx (voir encadré 1).

Le Blitter fonctionne à l'aide de trois sources (A, B et C) et d'une destination (D). En pratique, cela signifie qu'il possède quatre pointeurs. On lui indique les valeurs de départ de ces pointeurs et la nature de l'opération à effectuer, en précisant, entre autres, quelle opération logique il doit opérer entre les sources pour obtenir la valeur de la destination ; il suffit ensuite de lui indiquer la taille du bloc sur lequel il doit travailler ce qui le fera démarrer. On peut n'utiliser que quelques-uns des quatre canaux disponibles : pour un scrolling simple comme celui qui illustre cet article, il suffit de valider la source A et la destination D, il copiera simplement ce qui est pointé par A à l'adresse pointée par D.

Le blitter permet de plus de décaler le bloc transféré au point près, il est équipé pour cela d'un "barrel shifter" qui lui permet de faire des rotations de bits avec une maîtrise sans égale. Si vous devez travailler sur un bloc plus petit que la largeur du playfield, il faudra utiliser les modulos sur les canaux : par exemple, pour copier un bloc de 160 points de large dans un playfield de 320, le modulo sur D sera de 20 octets. Enfin, toujours pour un transfert simple, si votre zone source et votre zone destination se recouvre, il peut arriver que vous ayez à opérer le transfert de bas en haut (au lieu de haut en bas), cette possibilité est prévue (comme toujours !), il suffit de l'indiquer dans BLTCON1, mais il faut alors indiquer les valeurs de FIN de transfert dans les registres de pointeurs.

Une opération pour laquelle le blitter est également prévu est l'animation de bobs à l'écran (bobs = Blitter Objects, à ne pas confondre avec de vulgaires sprites qui sont des fantômes sans consistance - quoique, certains occultistes prétendent le contraire... qui croire ?), car contrairement au simple scrolling, l'affichage de bobs est une opération complexe ; il faut en effet copier le bob proprement sur une image, ce qui signifie qu'il doit y avoir utilisation de masques binaires. En effet, si l'on copiait directement le bob, la précision de 16 points (car on copie des mots de 16 bits) serait la cause de l'apparition de parasites sur les bords. Pour éviter cela, il faut utiliser les trois sources ; la première, A, pointe sur un masque du bob (le masque est simplement un OU inclusif logique entre tous les bitplans), la seconde, B, pointe sur l'image du bob, et enfin la troisième, C, pointe sur l'image où l'on va transférer le bob. L'opération logique sera (B AND A) OR (C AND NOT(A)) : le B AND A éliminera les parasites d'image à droite et à gauche (en effet, notre image de bob peut se trouver dans une table, juste à côté d'autres images, or l'on est obligé de transférer 16 points de plus que la taille de l'image, cela afin de faire le décalage, ce défaut se retrouvant pour le masque) ; le C AND NOT(A) annulera la mémoire vidéo à l'endroit où va se placer l'image du bob, et enfin le OR central opère réellement la copie. On fera de plus un décalage de bits sur A et B (les seuls d'ailleurs à pouvoir être décalés) pour positionner le bob au point près, et l'on effacera les résidus du masque sur les côtés en utilisant les masques sur A (BLTALWM & BLTAFWM, voir les registres).

Le listing ci-joint se contente de faire scroller un texte en 8 couleurs. Il faut pour cela avoir un jeu de caractères 32x32 8 couleurs, rangé par ordre alphabétique (A..J sur la première ligne, K..T sur la seconde, etc.) sur une image 320x256 décompressée et au format RAW (bitplans qui

LES REGISTRES DU BLITTER

```
BLTAPT.l ($50) Ces quatre registres sont
BLTBPT.l ($4C) les pointeurs sur les 3
BLTCPT.l ($48) sources (A, B et C) et la
BLTCPT.l ($54) destination dutransfert (D)

BLTAMOD.w ($64) Ces registres contiennent
BLTBMOD.w ($62) les modulos du transfert,
BLTCMOD.w ($60) cette valeur est ajoutée
BLTDMOD.w ($66) à la fin de chaque ligne.

BLTALWM.w ($44) Masque des 16pts à droite
                pour la source A
BLTAFWM.w ($46) Masque à gauche

BLTCON0.w ($40)
Bits 0-7 Sélection des fonctions logiques
         (voir encadré 2)
       8 USED Validation destination D
       9 USEC Validation source C
      10 USEB Validation source B
      11 USEA Validation source A
12-15 ASH0-ASH3 Décalage source A en pts

BLTCON1.w  ($42)
Bits 0 LINE tracé de ligne
           (0=Non (mode blocs) 1=Oui)
     1 DESC balayage des données
           (0=croissant 1=décroissant)
     2 FCI Mode de remplissage
           (0=Intérieur 1=Extérieur)
     3 IFE Mode de remplissage inclusif
     4 EFE Mode de remplissage exclusif
  5-11 Inutilisés
12-15 BSH0-BSH3 Décalage source B en pts

BLTSIZE.w ($58) Taille du bloc et
                démarrage du blitter
 Bits 0-5 Largeur du bloc en mots
          (64 mots=1024 pts)
     6-15 Hauteur en pts (1024 pts)
Écrire dans ce registre déclenche
          le transfert du bloc

DMACON.w ($96w $02r) Contrôle du DMA
Bits 6 Accès blitter au DMA (0=Non 1=Oui)
    10 Priorité blitter sur 68000 pour DMA
    13 Détection de bloc vide
       (0=Pas vide 1=vide)
    14 Blitter occupé (0=Non 1=Oui)

Sa programmation se fait dans cet ordre :
1 Contrôle du blitter BLTCON0
2 idem avec BLTCON1 (pour des transferts
de blocs, les bits 0, 2, 3 et 4 sont à 0)
3 Modulos BLTxMOD (souvent à 0)
4 Masques BLTAxWM (" à $ffff) pour source A
5 Pointeurs d'adrs BLTxPT source et dest.
6 Démarrage: indiquer taille bloc ds BLTSIZE
7 On peut attendre la fin du transfert en
  testant le bit 14 de DMACON
```

se suivent, contrairement au format IFF), et elle sera incorporée au programme par un INCBIN.

Vous pourrez facilement modifier des paramètres tels que le nombre de bitplans ou la taille du jeu de caractères. Il faudra simplement, en plus des valeurs des labels, modifier dans ce cas la routine qui recopie à l'écran l'image de la lettre, car étant faite au 68000 et optimisée, elle n'est pas paramétrable (désolé...). Le fonctionnement de ce programme est simple, à chaque IRQ (synchronisé avec le balayage), on décale une zone rectangulaire du playfield d'un point vers la gauche (pour cela, comme le décalage au point près du blitter ne peut se faire qu'à droite, on transfère 2 octets plus à gauche, puis on décale de 15 points à droite), après un nombre de décalages correspondant à la largeur de la lettre, on affiche une nouvelle lettre (on prend cette lettre dans l'image après avoir trouvé dans une table son adresse en fonction de son code ASCII). Le scrolling d'une partie du playfield se fait en décalant successivement la partie correspondante dans chaque bitplan. On peut optimiser ce programme en "entrelaçant" les bitplans, c'est-à-dire en mettant les unes à la suite des autres les lignes du bitplan 1, du bitplan 2, etc. L'affichage se fait alors en indiquant un modulo en fin de ligne. Si le playfield est bâti ainsi, le scrolling se fera avec un seul transfert blitter (il suffira alors de transférer une zone autant de fois plus haute qu'il y a de bitplans). Cette méthode est souvent utilisée pour l'affichage de bobs, car le démarrage du blitter représente une importante partie du temps machine, donc si l'on ne fait qu'un transfert (même si, au bout du compte, la quantité de données déplacées est identique), le gain est substantiel.

Nous avons donc fait rapidement le tour des puces graphiques de l'Amiga. Il faut signaler qu'en plus des transferts de blocs, le Blitter peut aussi tracer des lignes, cette possiblité, fort complexe est plus difficile d'accès et dépourvue d'intérêt sans une garniture telle que des routines d'affichage 3D, nous n'en avons donc dit mot ici. Pour finir cette série, nous verrons prochainement le listing complet d'une petite démo, scrolling multitrajectoire avec rotation et grossissement des lettres. D'ici là, bonne programmation.

François Fleuret

LES FONCTIONS LOGIQUES DU BLITTER

Il existe en mathématique ce que l'on appelle l'algèbre de Boole, dont on doit retenir, pour l'utilisation courante deux règles de calcul extrêmement puissantes; la commutativité du OR et du AND ainsi que la distributivité de ce dernier sur le premier (A AND (B OR C) = (A AND B) OR (A AND C)), grâce à ces règles de calcul, on peut ramener n'importe quelle opération logique entre A, B et C à des OR entre toutes les combinaisons de AND possibles entre les trois sources et leurs inverses logiques (A AND B AND C, A AND B AND NOT(C), ...). Nous noterons (comme cela est fait couramment) le AND de manière multiplicative et le OR de manière additive (comme les règles de calcul sont les mêmes, cela permet de faire des calculs booléens sans s'en rendre compte...), et nous noterons en minuscule le NOT d'une valeur. Donc, les bits 0 à 7 permettent de spécifier entre quelles combinaisons se font les OU.

BLTCON0:

Bits	Opération	
0	ABC	A AND B AND C
1	ABc	A AND B AND NOT(C)
2	AbC	A AND NOT(B) AND C
3	Abc	A AND NOT(B) AND NOT(C)
4	aBC	NOT(A) AND B AND C
5	aBc	NOT(A) AND B AND NOT(C)
6	abC	NOT(A) AND NOT(B) AND C
7	abc	NOT(A) AND NOT(B) AND NOT(C)

Pour utiliser ce tableau, il faudra ramener l'opération envisagée à un OU entre des combinaisons de AND. Par exemple, si l'on veut le fameux (B AND A) OR (C AND NOT(A)). On fera :

```
BA + Ca = BA(C+c) + Ca(B+b)
        = BAC + BAc + CaB + Cab
        = ABC + ABc + aBC + abC
Donc: bits 7,6,3 et 1 de BLTCON0, soit $CA.
```

Pour faire un scrolling simple, on aura l'opération A (seule !)

```
donc :    A = A(B+b) (C+c)
            = ABC + ABc + AbC + Abc
Donc, les bits 7, 6, 5 et 4, soit $F0
```

un ==> indique que la ligne de programme se poursuit sur la ligne suivante.

```
;     Scrolling d'un texte 8 couleurs
;     STMag François Fleuret
      opt d+
      include 'include/hardware/custom.i'
      include 'include/hardware/dmabits.i'
      include 'include/hardware/intbits.i'
      include 'include/hardware/cia.i'
      include 'include/exec/exec_lib.i'

CustomBase = $dff000 ; base coprocesseurs
Cia_AB     = $bfe001 ; base des Cia
Cia_BBase  = $bfd000

IRQvector  = $6c ; adresse du vecteur IRQ

CLR_ALL    = $7fff  ; valeur pour vider
                    ; dmacon ou intena
ScreenWidth = 320 ; 320 pts de large
ScreenHeight= 256 ; 256 pts de haut
ScreenDepth = 3   ; 4 bitplanes
ScreenModulo= 4   ; modulo de 32 points
ScreenMode  = $200+(ScreenDepth<<12)
                    ;Color on
;Calcul de valeurs qui serviront dans Inits.

Diwstrtyv = 172-ScreenHeight/2
Diwstrtxv = 289-ScreenWidth/2
Diwstopyv = ScreenHeight/2-84
Diwstopxv = 33+ScreenWidth/2
Ddfstrtv  = (Diwstrtxv-17)/2
Ddfstopv  = Ddfstrtv+(ScreenWidth/16-1)*8

Buffers  = 1    ; et une seule ram-vidéo

pr_MsgPort      =      $5C
pr_CLI          =      $AC

LineSize    =  ScreenWidth/8+ScreenModulo
PlaneSize   =  ScreenHeight*LineSize
BufferSize  =  ScreenDepth*PlaneSize
ScreenSize  =  Buffers*BufferSize

        section Principale,code

main:
  bsr   save_all      ; sauve valeurs Hard
        bsr    InitTable  ; Reloge table
        bsr    Inits   ;Initialise écran
        bsr    WaitClic  ;attend Mickey.
        bsr    restore_all;remet le hard
        moveq  #0,d0
        rts

; ***********************************
; Routine du scrolling proprement dit
; ***********************************

scrollY      = 64 ; ordonnée du scrolling
scrollSpeed  =  4 ; vitesse du scrolling
LrgFont =        32  ; largeur des lettres
HtrFont =        32  ; hauteur des lettres

LrgImgFont   = 320 ;largeur de l'image
HtrImgFont   = 256 ;des lettres
SizeBtplFont = LrgImgFont/8*HtrImgFont

scroll:
 lea   bitplanes,a0
 add.l #scrollY*LineSize,a0
 bsr   scrollbp ; premier bitplan
 add.l #PlaneSize,a0
 bsr   scrollbp ; second bitplan
 add.l #PlaneSize,a0
 bsr   scrollbp ; troisième bitplan
 rts

scrollbp:
 lea      CustomBase,a6
 move.l   a0,d0
 move.l   d0,bltapt(a6)  ; adr. de source
 subq.l   #2,d0
 move.l   d0,bltdpt(a6)  ; adr. de dest.
 clr.w    bltamod(a6)    ; pas de modulos
 clr.w    bltdmod(a6)
;move.w   #2,bltamod(a6) ; pas de modulos
;move.w   #2,bltdmod(a6)
 move.w   #$ffff,bltalwm(a6) ; on ne vire
 move.w   #$ffff,bltafwm(a6) ; pas de pts.
; utilise A et D, opération logique A seule
; et décalage de A de 15 points à gauche.
 move.w #(16-scrollSpeed)<<12+$9f0,bltcon0(a6)
 clr.w  bltcon1(a6) ; transfert croissant
 move.w #HtrFont*64+LineSize/2,bltsize(a6)
 move.w #HtrFont*64+LineSize/2-1,bltsize(a6)

BlitterBusy:
        btst     #DMAB_BLTDONE,dmaconr(a6)
        bne      BlitterBusy ;a-t-il fini ?

        rts

; ***************************************
; Routine relogeant la table des adresses
; ***************************************

InitTable:
        lea      AdrLettres,a0
        move.w   #25,d0
        move.l   #ImageFont,d1
BoucleInitTable:
        add.l    d1,(a0)+
        dbf      d0,BoucleInitTable
        rts

; *************************************
; Routine qui attend clic gauche souris
; *************************************

WaitClick:       ; Attend un clic souris.
  btst   #CIAB_GAMEPORT0,Cia_ABase+ciapra
  bne    WaitClick ; (ici 0=oui et 1=non)
        rts

; ******************************
; Routine qui initialise le hard
; ******************************

Inits:
        move.l  #DebutTexte,PointeurTexte

        move.w  #ScreenSize/12-1,d0
        lea     bitplanes,a0
        move.l  a0,a1
        add.l   #PlaneSize,a1
        move.l  a1,a2
        add.l   #PlaneSize,a2
        moveq   #-1,d1
clear:
        move.l  d1,(a0)+
        move.l  d1,(a1)+
        move.l  d1,(a2)+
        dbf     d0,clear

; Initialise les registres écran, le DMA,
; les IT et le pointeur copper.

        lea     CustomBase,a6
        move.w  #CLR_ALL,dmacon(a6)
                ; Vide DMACON.
        move.w  #CLR_ALL,intena(a6)
                ; Vide INTENA.

        move.w #ScreenMode,bplcon0(a6)
        clr.w  bplcon1(a6)
        clr.w  bplcon2(a6)
        move.w #ScreenModulo,bpl1mod(a6)
        move.w #ScreenModulo,bpl2mod(a6)
        move.w #(Diwstrtyv<<8)+ ==>
        Diwstrtxv,diwstrt(a6)
        move.w #(Diwstopyv<<8)+ ==>
        Diwstopxv,diwstop(a6)
        move.w #Ddfstrtv,ddfstrt(a6)
        move.w #Ddfstopv,ddfstop(a6)

        lea     color(a6),a0 ;met palette
        lea     palette(pc),a1
        move.w  #15,d0
metpalette:
        move.w  (a1)+,(a0)+
        dbf     d0,metpalette

        lea    irq(pc),a0 ;Adr routine IRQ
        move.l a0,IRQvector ;Vecteur IRQ.

; IT vbl uniquement.
     move.w  #INTF_SETCLR!INTF_INTEN! ==>
             INTF_VERTB,intena(a6)
; DMA bitplanes.
     move.w  #DMAF_SETCLR!DMAF_MASTER!==>
             DMAF_BLITTER!DMAF_RASTER,==>
             dmacon(a6)
     rts
palette:
```

```
        dc.w    ==>
 $000,$eef,$ddf,$ccf,$bbf,$aaf,$99f,$88f
        dc.w    ==>
 $77f,$66f,$55f,$44f,$33f,$22f,$11f,$00f

;       ** Routine de l'IT vbl **
irq:
        movem.l d0-a6,-(sp)
        lea     CustomBase,a6
    ; remet les pointeurs DMA bitplanes
        lea     bitplanes,a0

        move.l  a0,bplpt(a6)    ; bpl1pt
        add.l   #PlaneSize,a0
        move.l  a0,bplpt+4(a6)  ; bpl2pt
        add.l   #PlaneSize,a0
        move.l  a0,bplpt+8(a6)  ; bpl3pt

        bsr     scroll

        sub.w   #1,PositionLettre
        bpl     PasNouvelleLettre

   move.w  #(LrgFont/scrollSpeed)-1, ==>
PositionLettre
        move.l  PointeurTexte,a0
        move.b  (a0)+,d0
        bne     PasFinTexte
        lea     DebutTexte,a0
        move.b  (a0)+,d0
PasFinTexte:
        move.l  a0,PointeurTexte
        cmp.b   #' ',d0
        bne     PasBlanc

        moveq   #0,d2

     lea bitplanes+scrollY*LineSize+ ==>
          LineSize-LrgFont/8,a1
        move.w  #HtrFont-1,d1
ClearNvlLettre:
k       set     0
        rept    ScreenDepth
        move.l  d2,PlaneSize*k(a1)
k       set     k+1
        endr
        add.l   #LineSize,a1
        dbf     d1,ClearNvlLettre
        bra     PasNouvelleLettre

PasBlanc:
        sub.b   #'A',d0
        and.w   #$ff,d0
        add.w   d0,d0
        add.w   d0,d0
        lea     AdrLettres,a0
        move.l  (a0,d0.w),a0

     lea bitplanes+scrollY*LineSize+ ==>
          LineSize-LrgFont/8,a1
        move.w  #HtrFont-1,d1
CopieNvlLettre:
k       set     0
        rept    ScreenDepth
        move.l  SizeBtplFont*k(a0), ==>
        PlaneSize*k(a1)
k       set     k+1
        endr
        add.l   #LrgImgFont/8,a0
        add.l   #LineSize,a1
        dbf     d1,CopieNvlLettre

PasNouvelleLettre:

        move.w  #INTF_VERTB,intreq(a6)
        movem.l (sp)+,d0-a6
        rte

AdrLettres:

cx      set     0
cy      set     0
        rept    32
        dc.l    cx*LrgFont/8+cy* ==>
        HtrFont*LrgImgFont/8
cx      set     cx+1
        ifeq    cx-LrgImgFont/LrgFont
cx      set     0
cy      set     cy+1
        endc
        endr

; ****************************************
; SAVE_ALL arrête drive, sauve vecteurIRQ
; et les valeurs de INTENA et de DMACON.
; ****************************************

save_all:
  move.b  #CIAF_DSKMOTOR!CIAF_DSKSIDE! ==>
          CIAF_DSKDIREC!CIAF_DSKSTEP, ==>
          Cia_BBase+ciaprb
  move.l  IRQvector,saveIRQVector
  move.w  CustomBase+intenar,saveintena
  move.w  CustomBase+dmaconr,savedmacon
  rts

; ****************************************
; RESTORE_ALL remet le vecteur IRQ et les
; valeurs de INTENA, DMACON et COP1LC.
; ****************************************

restore_all:
      move.w  #CLR_ALL,CustomBase+intena
      move.w  saveintena,d0
      or.w    #INTF_SETCLR!INTF_INTEN,d0
      move.l  saveIRQVector,IRQvector
      move.w  d0,CustomBase+intena

      move.w  #CLR_ALL,CustomBase+dmacon
      move.w  savedmacon,d0
      or.w    #DMAF_SETCLR!DMAF_MASTER,d0
      move.w  d0,CustomBase+dmacon

        move.l  (_SysBase).w,a6
        lea     GraphicName,a1
        moveq   #0,d0
        jsr     _LVOOpenLibrary(a6)
        move.l  d0,a0
 move.l  38(a0),CustomBase+cop1lc;COP1LC
 clr.w   CustomBase+copjmp1     ;COPJMP1
        move.l  d0,a1
        jsr     _LVOCloseLibrary(a6)

        rts

saveintena:     ds.w    1
savedmacon:     ds.w    1
saveIRQVector:  ds.l    1

GraphicName:  dc.b  "graphics.library",0
        even

        section donnees,data

DebutTexte:
dc.b "ABC DEF GHI JKL MNO PQR STU VWX YZ "
dc.b "VOICI CE QUI EST POSSIBLE AVEC LE "
dc.b    "BLITTER ",0
        even

    section Font,data_c
ImageFont:
  incbin  'dh1:divers/pictures/font.dat'

  section MemVideo,bss_c ;section en Chip
bitplanes:
    ds.l    ScreenSize/4

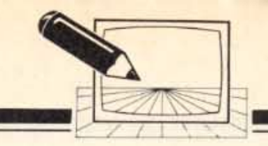
    section divers,bss

PositionLettre: ds.w    1
PointeurTexte:  ds.l    1

    end
```

ERRATUM

Dans les deux précédents listings de cette série, il y avait une petite erreur qui ne plantait que très rarement l'ordinateur, il s'agit du DC.B "graphics.lib" de RestoreAll auquel il manque le 0 de fin de chaîne, il fallait mettre DC.B "graphics.lib",0 pour que cela fonctionne parfaitement. Lorsque ce 0 est absent, dans la majorité des cas il y aura heureusement un 0 après la chaîne (celle-ci étant en fin de section) mais il arrivera que ce 0 soit absent, dans ce cas le OpenLibrary échoue et le CloseLibrary se fera sur rien, je vous laisse imaginer comment votre Amiga préféré encaisse ce genre de chose... Toutes nos excuses !



(Suite de la page 32)

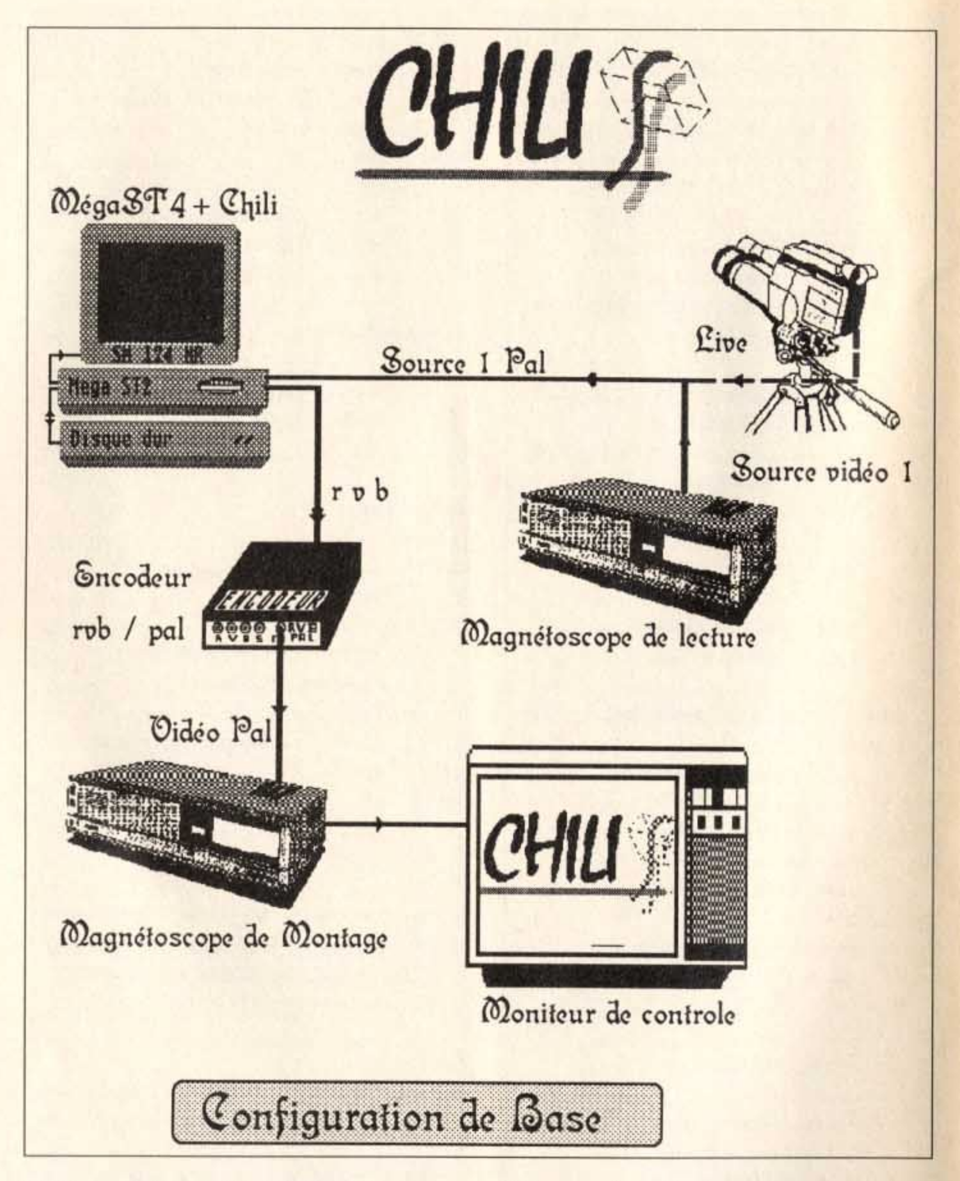
bloc moteur, ou encore la mise en oeuvre du catalogue de la collection d'été d'un styliste de mode ; il suffit d'un peu d'imagination !

g) Chili est un générateur d'effets torturant l'image sous tous ses angles : effets de rouleau, rideau, lamelles, mosaïque, stroboscope, etc. A noter que ces effets agissent sur l'image en temps réel.

Nous reviendrons, dès nos prochains épisodes, sur les applications logicielles accompagnant la carte Chili ainsi que sur un exemple de réalisation pratique. Nous nous préoccupons ce mois-ci de la partie "Installation" et description "Hardware".

INSTALLATION

La carte, de dimensions nobles (29.5 x 14.5 cm), regroupe un nombre impressionnant de composants (GALs, Processeurs vidéo, Ram, ...) et vient s'enficher dans le connecteur du bus d'extension du Mega ST. Il est à signaler que si vous possédez un quelconque montage (extension mémoire, interface genlock,PC Speed ou autres), il convient de vérifier que rien ne gêne la mise en place de la carte ; le côté composants se trouve en effet tourné vers le bas, sans accessoires hard, cela rentre donc très juste avec très peu d'espace, en raison de l'épaisseur des composants et connecteurs. L'alimentation s'enfiche dans l'emplacement libre derrière le lecteur de disquettes. Pour des raisons d'échauffement, on trouve un ventilateur fixé sur la carte, servant à maintenir une température constante. Un doute est permis sur la robustesse de l'alimentation du Mega ST, en raison de la forte consommation de Chili: 1.6 A en 5 volts et 0.3 A en 12 volts. Si vous possédez un Mega ST2 ou 4 d'origine équipé du ventilateur d'alimentation, vous ne risquez rien. Ayant pour ma part étendu mon Mega ST1 à 4 Mégas sans ventilateur d'origine, j'ai préféré utiliser une alimentation externe "solide" pour approvisionner le Mega ST4 et Chili, en raison de l'échauffement de l'alimentation d'origine. Avant de fixer la carte, il vous faut la configurer avec le type de moniteur que vous aller utiliser, dont voici une liste non exhaustive :

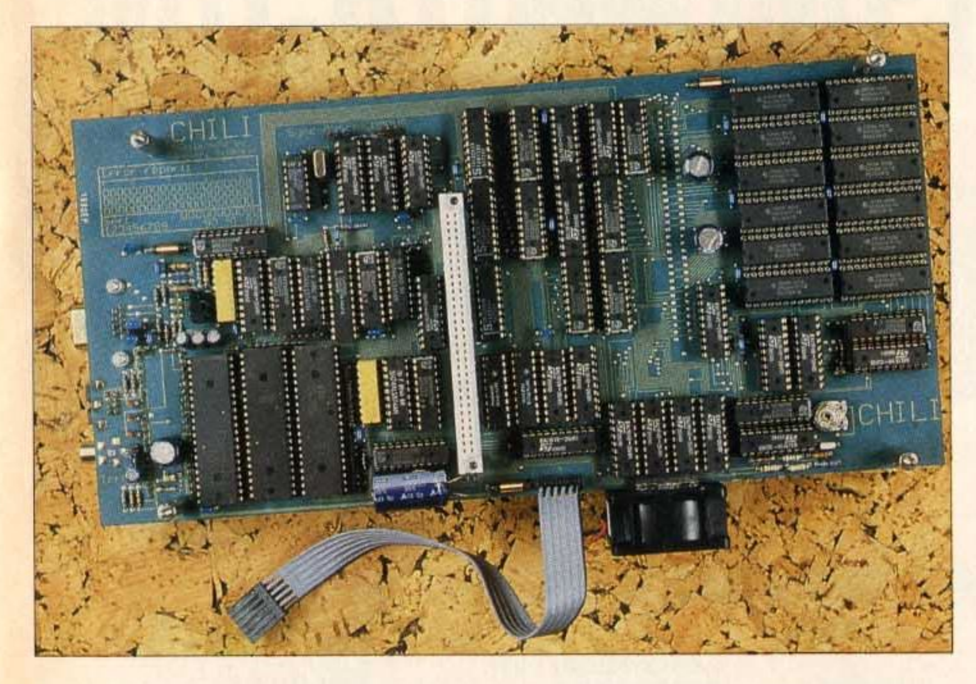


NEC Multisync II, EIZO Flexscan 9060, SONY KX-14CP1, Téléviseurs munis de la prise PERITEL, ATARI SC 1224, Encodeur RVB HAMA, Encodeurs Divers RVB-PAL.

3 connecteurs sont à votre disposition : un DB9 pour la sortie vidéo en composantes RVB uniquement, un connecteur Cinch pour l'entrée vidéo (1) composite au Standard PAL, et un connecteur mini DIN 8 broches pour une deuxième source vidéo extérieure (2) codée en RVB, qui doit de plus être synchronisée par rapport à la source principale, soit la vidéo composite qui rentre sur la Cinch en (1) ; je vous dévoilerai plus tard les astuces pour effectuer le raccordement de 2 sources vidéo pour le mixage.
Remarque importante et non négligeable pour le "budget" (la carte Chili se situant elle-même aux alentours de 16.000 F) : les câbles d'interconnections ne sont pas fournis (sauf le câble d'alimentation interne au Mega ST2), pratique courante sur la plupart des appareils vidéo. Dommage, car il vous faudra vous pencher sur le manuel pour en extraire le schéma de câblage des différentes prises, puis partir à la recherche du connecteur adéquat en fonction de vos périphériques (caméras/ moniteurs). Cela se terminera par la conception de votre câble introuvable auprès d'un revendeur sympa, qui ne manquera pas de vous faire constater le prix de ce type de connectique.

HARDWARE

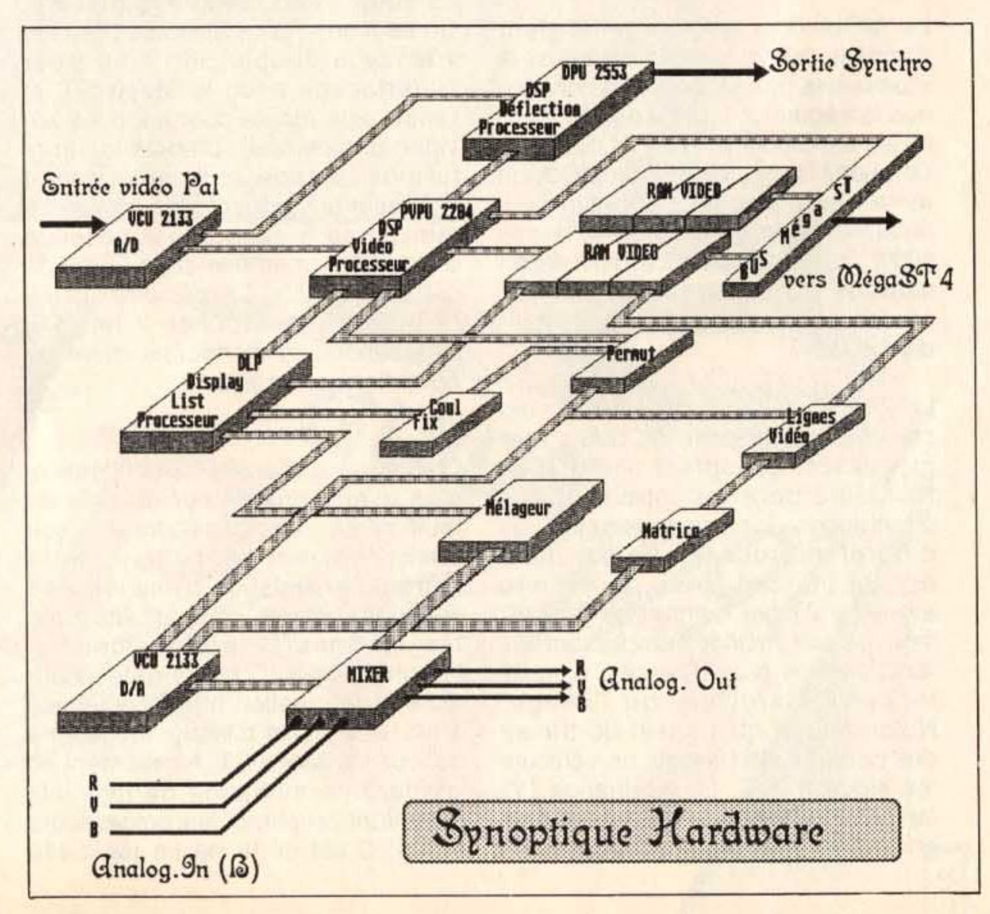
La carte Chili intègre un nombre impressionnant de circuits intégrés,

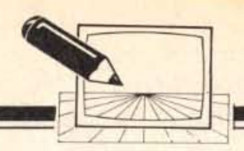


dont certains sont très spécialisés dans le domaine de la vidéo. La conception du circuit imprimé est remarquable et ne manque pas de professionnalisme ; c'est un circuit multicouche (4), comprenant les plans de masse propres à toute carte vidéo. Tous les composants actifs sont montés sur des supports dont certains sont du type "tulipe" pour les RAMs rapides à double port. Les connecteurs de raccordement vidéo externes sont au nombre de trois, un support libre sur la face (côté soudures) servira pour les extensions futures dont nous vous donnerons le détail à la fin de cet article. Etant donné la complexité de la technologie utilisée, il serait dommage de passer à côté de certaines explications sur la conception théorique de la carte. Je vais donc passer en revue la liste et les fonctions des différents processeurs utilisés.

Les processeurs sont issus de la technologie DIGIT 2000 développée par ITT Multicomposants, qui est une Famille très récente de circuits VLSI pour le traitement numérique de l'image vidéo. Ils sont principalement utilisés dans la télévision numérique pour le décodage des émissions par satellite en D2-MAC, et dans certaines techniques de codage/ décodage des émissions de télévision à péage. Ces circuits remplacent plusieurs centaines de composants dans les fonctions de contrôle, vidéo, déviation, et mettent en oeuvre les innovations suivantes: bruit infime, réduction de la taille du circuit imprimé équivalent, pas de vieillissement des circuits, pas de dérive ni de tolérance, en apportant une grande flexibilité par l'ouverture à la programmation externe par interface BUS. Ici, les processeurs sont interfacés au Bus du 68000 et pilotés par logiciel ; enfin, chose très importante, ils sont compatibles avec toutes les normes vidéo actuelles (SECAM/ PAL/ NTSC/ D2 MAC). Inutile de vous préciser que vous aurez du mal à vous procurer ces composants à la boutique spécialisée du coin. Renseignements pris chez ITT, ayant réussi par miracle à obtenir la doc technique propre à chaque processeur de la marque, je peux vous confirmer que ces processeurs sont distribués au compte goutte et par quantités d'environ 500 pièces minimum. Voilà, cela étant, nous allons passer au détail de la carte CHILI. Pour une meilleure compréhension, veuillez vous reporter au schéma du synoptique général de la carte Chili.

En premier lieu, on trouve le VCU 2133 un convertisseur Analogique-Digital sur 8 bits et un Digital-analogique sur 10 bits. Chaque convertisseur peut travailler à une fréquence d'horloge différente, ce qui introduit la possibilité d'effets spéciaux. Matrice RVB amplifiée et clampée (mise à niveau de la référence du noir du signal). De ce fait, en sortie de la carte, nous avons des signaux RVB qui pourront attaquer un moni-





teur quelconque. De plus, par l'ajout d'un encodeur RVB/PAL, on pourra enregistrer sur magnétoscope les images et les animations ainsi créées. Je vous proposerai prochainement la conception à peu de frais d'un encodeur RVB/PAL de bonne qualité, vous permettant l'enregistrement sur magnétoscope de toute image RVB, notamment les animations créées sur un ST (STF, STE, MegaST) sans n'avoir besoin d'aucune carte supplémentaire...

Vient ensuite le DPU 2553 qui est un processeur de balayage, ce qui explique la multitude de moniteurs raccordables à la carte. Il s'occupe de générer le balayage horizontal et vertical. On retrouve ses signaux directement en sortie vidéo sur le même connecteur que la sortie RVB. Lui aussi est pilotable par logiciel.
Le PVPU 2204 (VSP) est le processeur vidéo par excellence. Il joue le rôle de pilier dans la chaîne et comporte entre autres : multiplexeur de données ; multiplicateur de contrastes et couleurs ; interface bus pour la communication externe, ici pilotée par le 68000 du Mega ST; décodeur PAL compensé...

Le MCU 2632 est un générateur d'horloge qui va servir à cadencer la vitesse des processeurs. On constate que la fréquence utilisée est de 17.73 MHz, correspondant au PAL/SECAM. Le quartz étant mitoyen du MCU, on remarquera l'emplacement libre pour un autre quartz qui sert pour la norme NTSC dont la fréquence d'utilisation sera de 14.31 MHz. Encore une fois, on constate la prestation multistandard de Chili.

Le DLP (Display List Processor) est constitué d'une série de GALs programmées, qui offrent une grande flexibilité pour les modifications éventuelles. L'interfaçage entre les différents processeurs est donc assuré par ces GALs au nombre élevé de 21 qui forment un processeur primitif hybride remplissant les fonctions les plus diverses, comme le Zoom Hardware de l'image. Notons aussi qu'au sein du transit des données de l'image, on véhicule les signaux Y/C. La luminance (Y) est codée sur 8 bits (256 niveaux de gris), ainsi que la Chrominance (C).

CARACTÉRISTIQUES TECHNIQUES

Résolutions graphiques:

Mémoire	Affichage	Fréquence	Genlock:
512x500	460x290	50 Hz	Oui
512x500	460x290	60 Hz	Non
1024x250	920x250	50 Hz	Oui
1024x250	920x250	60 Hz	Non
512x500	460x500	50 Hz	Oui
512x500	460x500	60 Hz	Non
640x400	640x400	60 Hz	Non

Fréquence des pixels : 17.7 MHz
Fréquence mode TV : 15.625 MHz (Horizontal)
Fréquence mode Multisynchrone : 30 KHz

Entrées : Vidéo composite PAL
Composantes RVB
Sorties : RVB+HD/VD

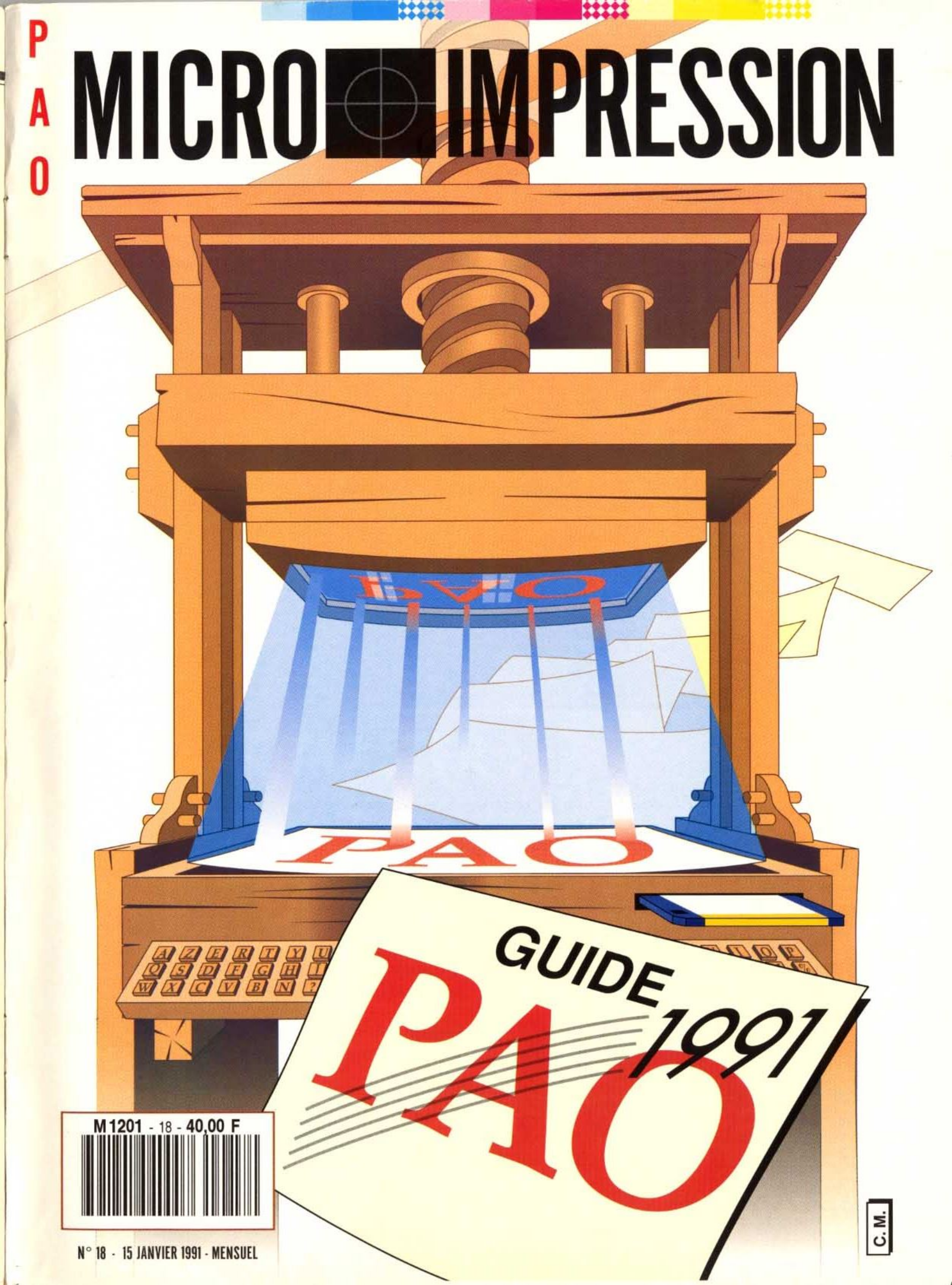
Zoom : 64 niveaux
Couleurs : 65 536 simultanées
Taille image : 256 000 octets Maxi
8 bits Luminance (256 niv. de gris)
8 bits Chrominance

La RAM vidéo comprend 384 Ko. Ce sont des Rams spéciales grande vitesse à double port : un pour l'interfaçage avec le Mega ST et l'autre plus rapide pour les données vidéo (temps réel). D'après la structure de la carte et l'emplacement vide pour un connecteur supplémentaire, il sera sans doute possible d'étendre la mémoire sous forme de carte enfichable. La mémoire actuelle permet de stocker 2 images simultanées et d'effectuer diverses opérations entre elles.

Chili augure, d'après sa conception, d'un avenir prometteur au sein du multimédia ; ses possibilités et son environnement permettent, entre autres, l'exportation d'images vers Retouche Pro, la compatibilité avec les scanners professionnels, l'impression sur imprimantes couleurs et la création de bibliothèques avec une qualité d'image inégalée à ce jour sur Mega ST. A cela vient se greffer une multitude de logiciels exploitant au mieux les processeurs vidéo. C'est cette partie logicielle que nous aborderons le mois prochain avec leurs qualités et défauts actuels, ainsi que les logiciels en cours de développement : le Raytracing, la Gestion du time code, la Gestion des images au format S-VHS et Y/C...

Henri ABDELOUAB

NDLR : Remercions ici Patrick de ALM pour son aide précieuse, ITT Multicomposants pour les docs techniques et le sus-nommé Henri pour sa passion pour la vidéo et l'informatique. En attendant le prochain épisode, regardez d'ores et déjà si vous ne possédez pas, sous une pile de vieux bouquins, des séquences vidéo de vos vacances au Népal, d'anniversaires loupés, de la première dent du chat, ou du dernier saut dans la piscine de grand-mère, qui dorment dans un coin sous la poussière et que vous pourriez métamorphoser avec un peu d'imagination et de doigté grâce à Chili...



PAO
MICRO IMPRESSION
PAO
PAO
A Z E R T Y U
Q S D F G H I
W X C V B N ?
GUIDE
PAO
1991
M 1201 - 18 - 40,00 F
N° 18 - 15 JANVIER 1991 - MENSUEL
C. M.

LA FORCE DU DESSIN

A l'époque du silex, le dessin était déjà un remarquable moyen d'expression et de communication.
Aujourd'hui, à l'âge du silicium, deux nouveaux outils ajoutent à votre système de micro-édition cette puissance naturelle qui accroît l'impact de tous vos documents : la force du dessin.

ARABESQUE PROFESSIONAL

Premier logiciel dédié à la création et au traitement simultané d'images vectorielles et bit-map, Arabesque Professional vous offre un confort d'utilisation, une puissance, et une rapidité d'exécution inégalables.

CONVECTOR

Il devient même inutile de dessiner, avec Convector, pour transformer instantanément un graphisme bit-map en un graphisme vectoriel parfait pour la micro-édition.
Complément autonome de tout logiciel de dessin, Convector devient idéal en tandem avec Arabesque Professional.

Arabesque, Convector, et Arabesque Professional sont compatibles Atari ST et TT.



Tél: (1) 43 44 78 88. Fax: (1) 43 44 90 96